Antonio A.G. Sa da Costa
John L. Wilson

A Numerical Model of Seawater Intrusion in Aquifers

MIT Sea Grant
College Program

Massachusetts
Institute of Technology
Cambridge
Massachusetts 02139

Report No.
MITSG 79-27
December 1979

A NUMERICAL MODEL OF SEAWATER INTRUSION IN AQUIFERS

by

Antonio A.G. Sa da Costa
and
John L. Wilson

Sea Grant College Program
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Report No. MITSG 79-27
Index No. 79-327-Ccj
December 1979

A NUMERICAL MODEL OF SEAWATER INTRUSION IN AQUIFERS

by

António A. G. Sá da Costa

and

John L. Wilson

ABSTRACT

Seawater intrusion along irregular coastlines and under offshore islands is a classical result of groundwater development. On a regional scale vertical effects, such as dispersion in the mixing zone between freshwater and seawater, and vertical flow are often negligible. Thus a two-layer horizontal flow model is appropriate. Freshwater is the upper layer, seawater is the lower layer, and an immiscible interface between the two is assumed. The development of a numerical model of this type is described: SWIM, a Sea Water Intrusion Model.

The finite element method with a Galerkin formulation is used to solve the vertically integrated governing equations (following the "hydraulic" or Dupuit approach). The model accounts for a variety of aquifer situations that occur in the field, such as: homogeneous or non-homogeneous and isotropic or anisotropic aquifer properties; leaky or non-leaky and phreatic or confined status; and constant or time varying boundary conditions. SWIM simulates situations in which the freshwater either floats over a seawater body, forming a *lens*, or lies over a finite extent seawater wedge, defining a seawater wedge *toe*. The toe movement is accurately represented using a fixed finite element mesh, and a non-linear description of the parameters inside those elements containing a toe. Numerical experiments were performed using the 1-D gravitational segregation problem to judge the sensitivity of the model, and particularly the toe tracking algorithm, to different parameters and options. Other applications and verifications of the SWIM code are also included: classical 1-D seawater intrusion, development of a freshwater lens over seawater, seawater intrusion in a leaky coastal aquifer, injection of freshwater into a saline aquifer and seawater intrusion toward a pumping well. SWIM has demonstrated through all of these examples that it is an accurate and efficient model of seawater intrusion in aquifers, whether freshwater occurs as a lens over seawater, or as a layer over a finite length seawater wedge.

ACKNOWLEDGMENTS

This study was primarily sponsored by the Sea Grant Office of the National Oceanic and Atmospheric Administration, Department of Commerce under OSP 85464, with additional support from the Agency for International Development through the MIT Technology Adaptation Program under OSP 85054. The computations were carried out at the MIT Information Processing Services.

The authors wish to thank Professor Klaus-Jurgen Bathe from the Department of Mechanical Engineering for his advice and especially for allowing the use of the ADINAT code (Bathe, 1977) as a starting point for the codification of SWIM. Discussions with Messrs. Pedro Restrepo and Lloyd Townley are greatly appreciated. Special thanks are due to Mr. Mohammad Khoshgoftaar from the Department of Mechanical Engineering for his help in getting the senior author acquainted with the ADINAT code.

The patience, skill and speed of Ms. Carole Solomon in typing the manuscript is sincerely appreciated.

This report is essentially a reproduction of the thesis by António Sá da Costa, submitted in October 1979, in partial fulfillment of the requirements of the degree of Doctor of Philosophy. Dr. John L. Wilson, Associate Professor of Civil Engineering, served as thesis advisor.

FOREWORD TO THIS REPORT

This report is the first of a series of publications which describe SWIM, a numerical code to simulate seawater intrusion in aquifers. Shortly a Users' Manual will follow.

Inquiries about SWIM and related publications should be addressed to:

Professor John L. Wilson
Room 48-209
Department of Civil Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139 U.S.A.

The Sea Grant Marine Resources Information Center maintains an inventory of all technical and advisory publications. We invite orders and inquiries to:

MIT Sea Grant Program
Marine Resources Information Center
77 Massachusetts Avenue
Building E38-302
Cambridge, MA 02139

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

A	area $[L^2]$
$\underline{\underline{B}}$	space derivative operator [1/L]
$\underline{\underline{\mathbf{B}}}$	space derivative of the interpolation functions [-]
b	saturated or aquifer thickness [L]
b'	leaky layer thickness [L]
b_e	extra-layer thickness [L]
$\underline{\underline{C}}$	global non-symmetric storage matrix [-]
$\underline{\underline{C}}^*$	global symmetric storage matrix [-]
ℓ	direction cosine vector [-]
$\underline{F}$	equilibrated flow $[L^3/T]$
$\underline{F}_o$	out of balance flow $[L^3/T]$
$\underline{\underline{H}}$	interpolation function matrix [-]
h_i	interpolation function components [-]
$\underline{\underline{J}}$	Jacobian matrix [-]
K	aquifer permeability [L/T]
$\underline{\underline{K}}$	total conductivity matrix $[L^2/T]$
K'	vertical permeability of the leaky layer [L/T]
$\underline{\underline{K}}'$	vertical permeability of the leaky layer matrix [L/T]
$\underline{\underline{K}}^*$	effective conductivity matrix $[L^2/T]$
K'_c	parameter of the coastal boundary [L/T]
$\underline{\underline{k}}$	permeability tensor [L/T]

$\underline{\underline{KL}}$ linear part of $\underline{\underline{K}}$ [L^2/T]

$\underline{\underline{KNL}}$ nonlinear part of $\underline{\underline{K}}$ [L^2/T]

L toe location [L]

L( ) general differential operator [1/T]

$\underline{\underline{L}}$ third type boundary condition (leakage) matrix [L^2/T]

N accretion rate [L/T]

n effective porosity [-]

$\underline{n}$ unit vector [-]

Q_c^f freshwater flow to the sea [L^2/T]

$\underline{Q}$ source/sink vector [L/T]

$\underline{\mathbf{Q}}$ specified flow vector [L^3/T]

q source/sink term [L/T]

$\underline{q}$ independent variable [L/T]

r local coordinate [-]

S elastic storativity or specific yield [-]

$\underline{\underline{S}}$ storage matrix [-]

$\underline{\underline{S}}^*$ symmetric storage matrix [-]

S_s specific storage [1/L]

s boundary segment [L]

s local coordinate [-]

$\underline{\underline{T}}$ transmissivity matrix [L^2/T]

$\underline{\underline{\mathbf{T}}}$ conductivity matrix [L^2/T]

t time [T]

v celerity of the toe [-]

$\underline{v}_s$ velocity of propagation of the free-surface [L/T]

$\underline{\underline{W}}$ weighting function matrix

w_i weighting function components

x, x_i coordinates (i=1,2) [L]

$\underline{Y}$ vector [L]

y coordinate [L]

z coordinate [L]

Superscripts

$( )^f$ freshwater quantity

$( )^s$ seawater quantity

$( )^T$ transpose matrix

Subscripts

$\underline{\underline{\ }}$ matrix quantity

$\underline{\ }$ vector quantity

$( )_b$ boundary quantity

$( )_c$ coastal boundary quantity

Greek Alphabet

α soil vertical compressibility [LT^2/M]

β water compressibility [LT^2/M]

γ specific weight [$M/L^2/T^2$]

δ Dirac delta function [-]

Δ differential [-]

$\Delta\gamma = \gamma^s - \gamma^f$ [$M/L^2/T^2$]

ζ interface depth below mean sea level [L]

λ eigenvalue [-]

ξ interface surface [-]

ρ specific mass [M/L^3]

τ dimensionless time [-]

$\underline{\Phi}$ state vector containing freshwater and seawater piezometric heads [L]

$\underline{\Phi}'$ vector of the piezometric heads in the adjacent aquifers [L]

ϕ piezometric head in the aquifer [L]

ϕ' piezometric head in the adjacent aquifers [L]

$\overline{\phi}$ average piezometric head [L]

∇ gradient operator

Chapter 1

GENERAL INTRODUCTION

The growth of population in coastal areas and on offshore islands, along with a corresponding growth of recreational, agricultural and industrial activities, has created an increasing demand for freshwater. Often this increase in demand is satisfied by a more intensive pumping of groundwater, upsetting the long term natural balance of freshwater flowing underground to the sea, and the heavier underlying wedge of seawater. As a consequence the seawater wedge advances inland, encroaching on the underground supply of freshwater and threatening its availability with contamination by brackish water.

1.1 Sea Water Intrusion Modeling for Aquifers

Physical, analytical and numerical models have been used to evaluate various groundwater management schemes, and to predict their effect on seawater intrusion. Physical models for seawater intrusion are primarily of the Hele-Shaw type, and are quite expensive to build and calibrate. Once built they represent only one particular location and situation. The use of this approach for applied modeling has decreased with the advent of high speed digital computers and the development of numerical models. Analytical models are used for feasibility studies and, along with physical models, serve as a term of comparison for the calibration and verification of numerical models. All three model types use some form of the governing differential equations for predicting the behavior of an aquifer. These equations, with the appropriate boundary and initial conditions, constitute a mathematical

statement which is a conceptualization of the field situation. Solutions of these models have been successfully applied to the evaluation of many coastal aquifer problems (e.g., see literature reviews in Chapter 2 and Bear, 1970, 1972 and 1979).

Analytical models are only able to represent simplified aquifer conditions: for example, simple geometry, a homogeneous and isotropic aquifer, constant head and flow boundary conditions, and in many cases, steady state flow. However, they can be quickly implemented, with no special requirements except a pencil and paper, and give a reasonable first order approximation for many situations.

Numerical methods, implemented on high speed computers, are more general in their applicability, and have been used to successfully obtain solutions for very complex aquifer situations (see, e.g., Shamir and Dagan, 1971; Pinder and Page, 1976; and reviews in Bear, 1979; Peaceman, 1977; and Pinder and Gray, 1977). A numerical model is defined here as a computer program that uses numerical methods to obtain an approximate solution to the mathematical model. Some of these numerical models, and the one presented here is no exception, have been designed for use by groundwater hydrologists and/or geohydrologists who possess only a limited numerical methods background. Yet, in spite of the best efforts of code writers to improve numerical methods so as to make the computer codes as reliable, efficient, and automatic as possible, the user of computer code is constantly faced with many decisions that are only code related and have little to do with the physical problem he really

wants to solve. For example, he must decide on numerical method (finite differences or finite elements), space discretization (grid size), time step, iteration parameters (iterative solutions only), etc. Only with experience are these decisions readily and confidently made by the model user.

1.2 Sea Water Intrusion Model (SWIM)

This report presents a finite element model named SWIM, an acronym for Sea Water Intrusion Model. SWIM simulates groundwater flow in coastal aquifers and under offshore inlands, in which freshwater and seawater are both present. The vertical integrated flow equations, sometimes called the hydraulic or Dupuit equations, are used to represent regional, essentially horizontal two-dimensional (2-D) flow. The aquifer can be homogeneous or non-homogeneous, isotropic or anisotropic, leaky or non-leaky, confined or phreatic. Time varying or constant boundary conditions as well as steady state or transient solutions are also accounted for in the code.

In most practical situations the transition zone between seawater and freshwater, where mixing and dispersion phenomena occur, is quite narrow when compared with the overall saturated thickness of the aquifer formation. In these cases an immiscible interface between seawater and freshwater is an appropriate and very convenient approximation (see, e.g., Chapter 2 and Chapter 5; Bear, 1970, 1972, 1979; Shamir and Dagan, 1971; Pinder and Page, 1976). This approximation leads to the definition of two types of freshwater aquifers: *lens*

when the freshwater layer "floats" on top of the seawater layer; *toe*, when the immiscible interface intersects the bottom of the aquifer, defining a seawater wedge "toe". SWIM, which utilizes the immiscible interface assumption, can simulate both types of aquifers. When a toe exists, a fixed mesh algorithm was developed to accurately and economically track the toe movement, especially when compared to traditional mesh regeneration models.

1.3 Scope

Chapter 2 reviews the state of the art of analytical and numerical seawater intrusion models. A classification of the different types of models is outlined. A review of research performed in related physical problems (e.g., petroleum reservoir simulation and the Stefan problem) is also presented. The chapter ends with the discussion of the approach selected for SWIM.

The numerical model is formulated and discussed in Chapter 3. A review of the hydraulic approach governing equations, based on a derivation that is presented in Appendix A, and its boundary conditions is given. Then the Galerkin finite element method is applied. A description of iterative techniques for handling equation non-linearities and the time integration technique follows.

Chapter 4 presents a brief discussion of different methods to track the toe, which is a moving "front" or "boundary". There is a detailed description of the proposed toe tracking algorithm selected for SWIM. A study of the sensitivity of this technique to different

factors, such as space and time discretization, is also presented. For this sensitivity analysis the 1-D porous media gravitational segregation problem was used.

To verify the applicability of SWIM to different types of situations involving two fluids, several hypothetical test cases were examined and are presented in Chapter 5. These include, besides the gravitational segregation problem mentioned above, classical 1-D Dupuit type seawater intrusion in aquifers, development of a lens of freshwater over seawater, intrusion in a leaky coastal aquifer, radial injection of freshwater in a saline aquifer, and seawater intrusion toward a coastal pumping well.

Finally Chapter 6 presents a summary, conclusions, and directions for future research.

Chapter 2

BACKGROUND AND LITERATURE REVIEW

2.1 Introduction

Some of the more significant contributions to the theoretical analysis of seawater intrusion are briefly reviewed in this chapter. A classification of the different approaches is presented. Contributions to the literature are summarized according to this classification in two tables, one for analytical models and the other for numerical models.

Other physical situations that can be related in some way to seawater intrusion in aquifers, such as the petroleum reservoir and Stefan type problems, are presented in the middle part of the chapter. These problems are related either through the physics of the problem, the method of solution, or both. The door remains open for future work to adapt the methodology presented in this report, and possibly even the SWIM code, to solve some of these related problems.

The chapter ends with a statement and discussion of the approach selected for SWIM, the Sea Water Intrusion Model developed in this report.

2.2 Seawater Intrusion Problems

2.2.1 Classification of the Models

In the literature there is no explicit systematization of the different approaches used to solve the problem of seawater intrusion. For a better definition and understanding of the capabilities of the

present model, SWIM, a classification of the previous work is necessary. The classification groups are briefly presented in Table 2.1, and the criteria used in the specification of the various categories are explained below.

The term *methodology* characterizes the "tools" used. Physical models of seawater intrusion are usually of the Hele-Shaw type (see e.g., Bear and Dagan, 1963, 1964a; Collins, Gelhar and Wilson, 1972; Collins and Gelhar, 1971; Mualem and Bear, 1974) and are only mentioned here for the sake of completeness. No further attempt to explore classification os this type of model is presented here. Analytical models and numerical models use differential equations to represent a given situation. The difference between them is the method used to solve these equations. Using several simplified assumptions, most analytical models lead to an exact closed form solution of the governing differential equations. However, this is not always possible. Among the more common simplifying assumptions are: simple geometry, homogeneity and isotropy of aquifer properties, steady state conditions, and linearization of the equations or boundary conditions. Solutions to analytical models usually require only pencil, paper and tables of common functions. Sometimes numerical integration is necessary, utilizing the services of a computer, but the numerical burden is slight.

Numerical models discretize the continuous time and space domain of the governing equations, resulting in a series of simultaneous algebraic equations which are usually solved by Gauss elimination or

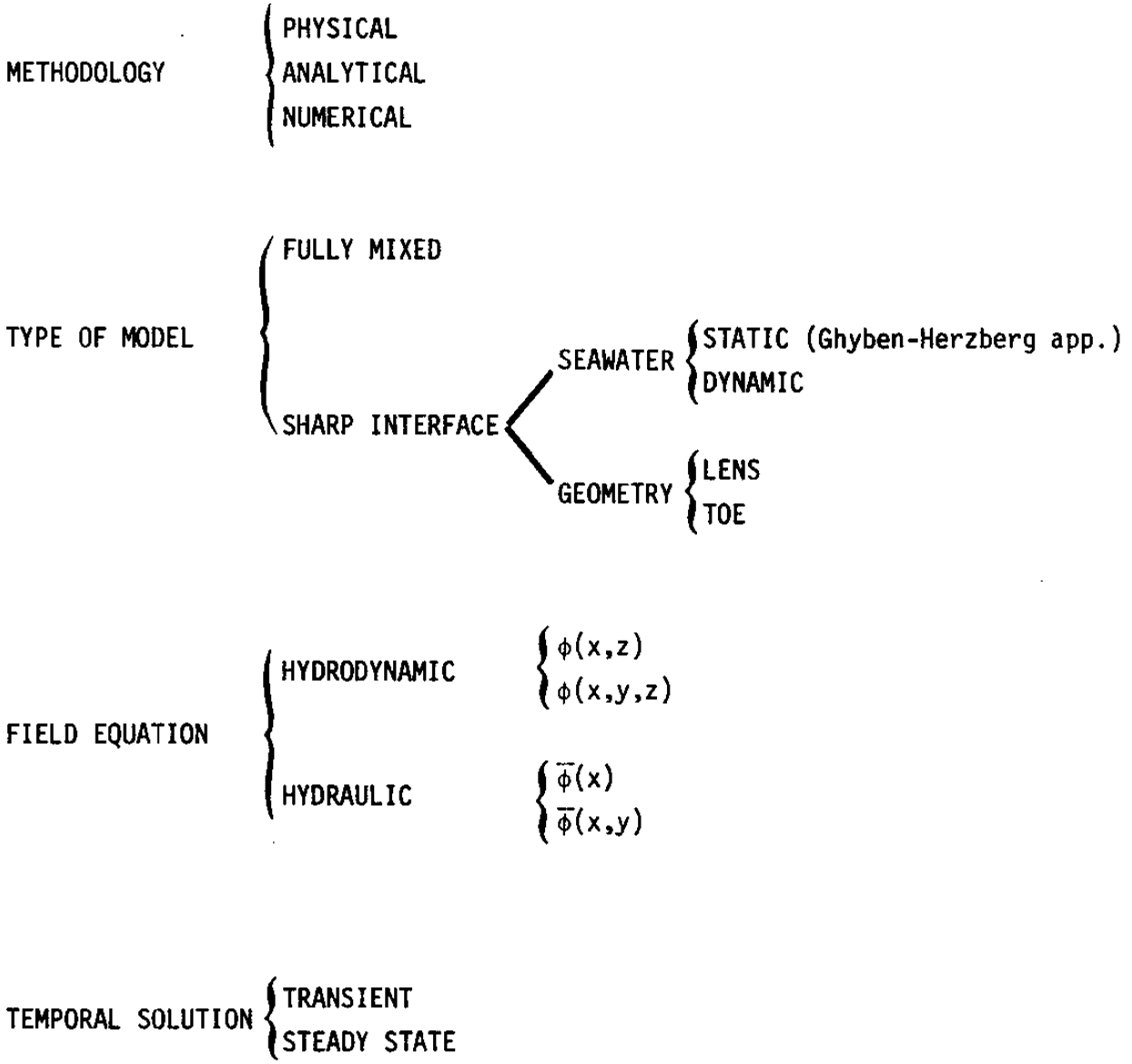
iteration. Although some simplification of the physical system is required, numerical models can generally be applied to a wide range of situations involving complicated geometry, heterogeneity, anisotropy, and steady or transient flow. Their use requires computer facilities, and for most applied problems, the prior development and documentation of a computer code.

Comparing the two methods, it is clear that analytical models provide an extremely useful first order approximation of a field problem, as well as the means to verify more versatile numerical models. Yet it is just as clear that most analysis in the future will rest on the use of numerical codes. The remaining part of the classification presented in Table 2.1 is applicable only to analytical and numerical models.

The term *type of model* concerns the way the transition zone between freshwater and seawater is treated. In reality, freshwater and seawater are two miscible fluids, with a transition zone between them. In this zone dispersive mixing occurs between the two fluids. Often this transition zone is very thin, compared with the aquifer thickness. When this occurs, an abrupt or immiscible interface between freshwater and seawater usually is assumed (Bear, 1972, 1979). Different equations are solved for different "types of models". A *fully mixed model* requires the coupling of two equations: the flow equation and the mass transport (or convective-dispersion) equation. A *sharp interface model* requires the solution of two flow equations: one for freshwater, the other for seawater. In some cases this last equation is dropped because

Table 2.1

Classification of Models of Seawater Intrusion in Aquifers



ϕ – piezometric head;

$\bar{\phi}$ – depth averaged piezometric head;

x,y,z – Cartesian coordinates (z vertical).

the seawater phase is assumed to be *static*. This is called the Ghyben-Herzberg approximation. When the seawater flow is accounted for, the model is designated as a *dynamic* seawater model. Another parallel classification for a sharp interface model concerns the capability of the model to analyze the intersection of the interface with the bottom of the aquifer. If the interface does not intersect the bottom of the aquifer, the freshwater body floats on top of the seawater and has the shape of an optical lens. If the interface intersects the bottom of the aquifer, at a point usually called the toe of the interface (see Figure 2.1), then the seawater zone is of limited lateral extent. A sharp interface model that can only simulate lens type situations is designated here by the term *lens* model; if it can handle situations in which a toe exists it is called a *toe* model.

The *field equation* describes the way vertical flow is handled. If vertical flow and the vertical coordinate, z, is retained in the equation of flow, then it is designated as a *hydrodynamic* model; if the equation of flow is vertically averaged, it is designated as a *hydraulic* model. The hydrodynamic equation can have up to three spatial variables, (x,y,z), although almost all hydrodynamic models of seawater intrusion involve only two spatial variables: the vertical direction z, plus one horizontal direction, x. The hydraulic equation can have up to two spatial variables, (x,y), but both must be in the horizontal plane. The *hydraulic approach* is synonymous with the designations "essentially horizontal flow" and/or "Dupuit approximation" (see, e.g., Bear, 1979).

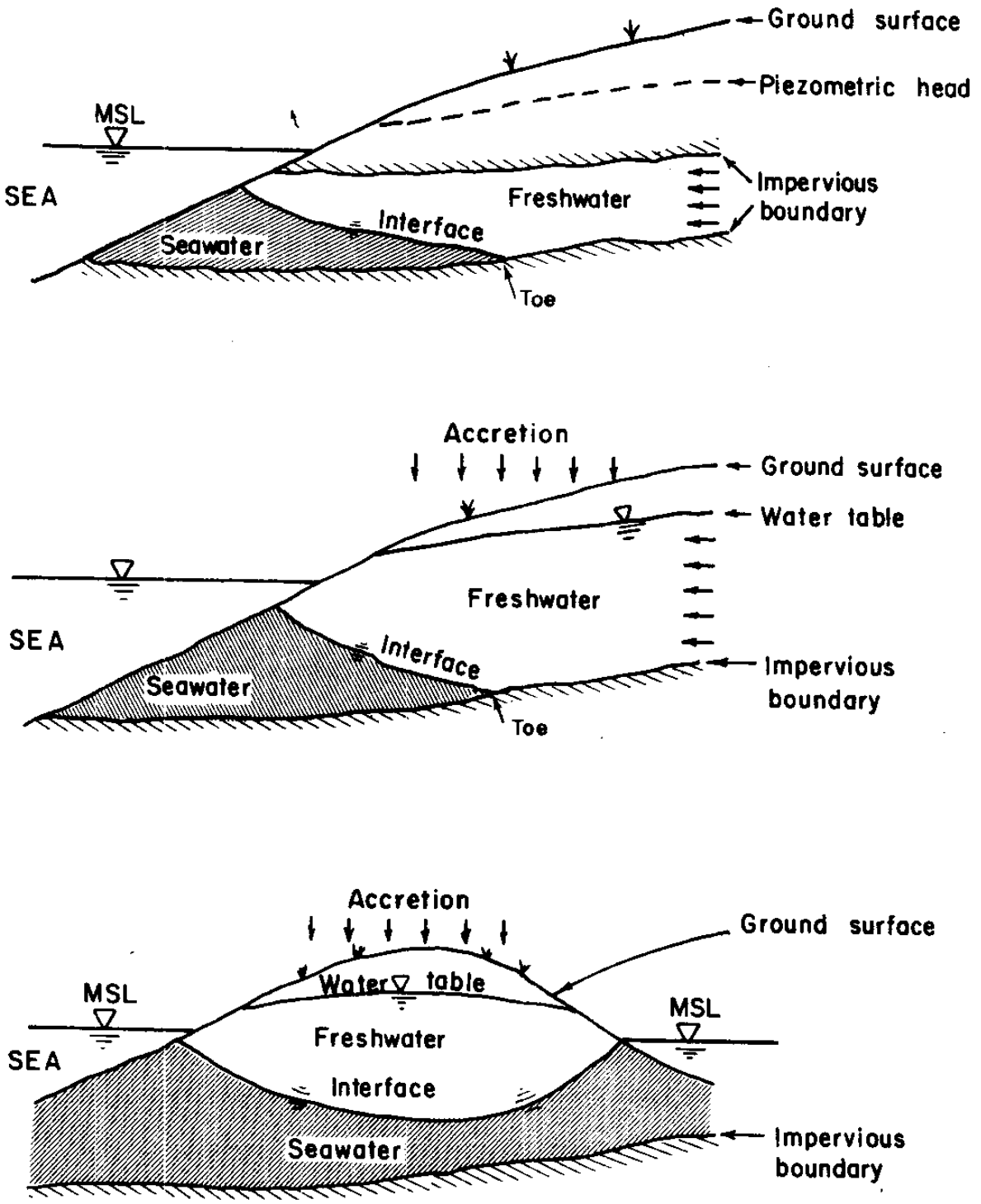


Figure 2.1 - Seawater Intrusion in Different Types of Aquifers

The term *temporal solution* concerns whether the model solves steady state and/or transient problems. If transient, the additional variable is time t.

Some of the more significant contributions found in the seawater intrusion literature are classified according to Table 2.1. The resulting classifications, one for analytical models and the other for numerical models, are presented as Tables at the beginning of the following two sections.

2.2.2 Analytical Models

The first studies reported in the literature describing seawater intrusion in aquifers are those of Drabbe and Ghyben in 1889 and Herzberg in 1901. Both papers focussed on the nothern coastal areas of Europe. They examined the analytical steady state balance between freshwater and seawater assuming that:

1- freshwater and seawater are separated by an abrupt or sharp interface,

2- seawater is in static equilibrium.

This last assumption is commonly known in the groundwater literature as the Ghyben-Herzberg approximation. Later, Hulbert (1940) used these two assumptions to conclude that the interface depth (ζ) below the mean sea level (MSL) is given by:

$$\zeta = \frac{\gamma^f}{\gamma^s - \gamma^f} \phi^f \qquad (2.1)$$

in which: γ^f freshwater specific weight,

γ^s seawater specific weight,

ϕ^f freshwater piezometric head.

Eq. 2.1 is often called the Ghyben-Herzberg equation.

The new era of the study of seawater intrusion in aquifers began with the work of Glover (1959), Henry (1964a,b), and Bear and Dagan (1964a). These and later contributions are classified in Table 2.2.

Glover (1959), Henry (1964a), Bear and Dagan (1964a) Rumer and Shiau (1968), Verruijt (1968) and Van der Veer (1977), accounting for vertical flow and using conformal mapping and/or hodograph techniques, calculated several solutions for the steady state position of a sharp interface in coastal aquifers, subject to different boundary conditions. These studies examined a vertical cross section, $\phi(x,z)$, of a homogeneous and isotropic aquifer. Glover (1959), Rumer and Shiau (1968), and Van der Veer (1977) looked at infinitely deep aquifers, so they are classified here as *lens* models. Although Bear and Dagan (1964a) also looked at a lens model, they and Henry (1964a) examined seawater intrusion in a finite depth confined aquifers, a *toe* situation. Rumer and Shiau (1968) included the effects of anisotropy and layered non-homogeneity. Rumer and Harleman (1963) performed laboratory experiments using a sand box physical model that confirmed this theoretical work. Henry (1964b) also solved the steady state problem of seawater intrusion in a confined aquifer, considering vertical flow and dispersion. His work is the only analytical study of seawater

Table 2.2

Classification of Analytical Models

TYPE OF MODEL		AUTHOR	INDEPENDENT VARIABLES	TYPE OF AQUIFER	METHOD OF SOLUTION	FIELD EQUATION
FULLY MIXED		Henry (1964 b)	$\phi^f(x,z);C(x,z)$	confined	Fourier/Galerkin	hydrodynamic
SHARP INTERFACE	LENS	Glover (1959)	$\phi^f(x,z)$	confined	conformal mapping	hydrodynamic ssw
		Van der Veer (1977)	$\phi^f(x,z)$	phreatic	potential flow	hydrodynamic ssw
		Rumer+Shiau (1968)	$\phi^f(x,z)$	confined anisotropic or layered	conformal mapping	hydrodynamic ssw
		Hantush (1968)	$\bar{\phi}^f(x,y,t)$	phreatic	approximated PDE	hydraulic ssw
	TOE	Henry (1964 a)	$\phi^f(x,z)$	confined/phreatic	conformal mapping	hydrodynamic ssw
		Bear+Dagan (1964 a)	$\phi^f(x,z)$	confined	hodograph method	hydrodynamic ssw
		Bear+Dagan (1964 b)	$\bar{\phi}^f(x,t);\zeta(x,t)$	confined	approximated PDE	hydraulic dsw
		Collins+Gelhar(1971)	$\bar{\phi}^f(x)$	leaky confined	approximated PDE	hydraulic ssw
		Mualem+Bear (1974)	$\bar{\phi}^f(x)$	multi-layered leaky	approximated PDE	hydraulic ssw
		Verou (1978) Hashish et al.(1979)	$\bar{\phi}^f(x)$	leaky confined	approximated PDE	hydraulic ssw
		Strack (1976)	$\bar{\phi}^f(x,y)$	confined/phreatic	potential flow	hydraulic ssw
		Kishi+Fukuo (1977)	$\bar{\phi}^f(r,\theta)$	confined	approximated PDE	hydraulic ssw
		Kashef (1975,1976) Kashef+Smith (1975)	$\bar{\phi}^f(x,y,t)$	confined	superposition	hydraulic ssw

C – concentration of salt ;
ϕ^f – freshwater piezometric head ;
$\bar{\phi}^f$ – depth averaged freshwater piezometric head ;
ζ – interface depth ;
x,y,z – Cartesian coordinates (z vertical) ;
r,θ – cylindrical coordinates ;
t – time ;
PDE – partial differential equation ;
ssw – static seawater (Ghyben-Herzberg approximation);
dsw – dynamic seawater.

intrusion examining the effect of mixing on the location and extent of the transiiton zone.

Bear and Dagan (1964b) analytically applied the hydraulic approach, and experimentally used a Hele-Shaw model, to study a confined aquifer with a sharp interface. The interface moved due to changes of incoming freshwater flow. They found two expressions for the interface, one for an advancing wedge and the other for a retreating wedge. Other basic applications of the sharp interface/hydraulic approach are given in Rumer and Harleman (1963) and Fetter (1972), as well as the reviews by Bear (1970, 1972 and 1979).

Using the hydraulic approach and the Ghyben-Herzberg sharp interface approximation, Hantush (1968) established and solved analytically differential equations that approximate different situations of the unsteady movement of a freshwater lens in infinitely deep, homogeneous and isotropic phreatic aquifers. By assuming an infinitely deep aquifer Hantush avoids the problem of the intersection of the interface with the bottom, that is the freshwater body floats as a lens over the seawater. This also permits the use of the Ghyben-Herzberg approximation.

Other authors have examined the steady state shape and position of the interface in leaky aquifers, using the sharp interface/hydraulic approach. Collins and Gelhar (1971) analyzed seawater intrusion in a leaky aquifer by solving the governing equations analytically inland of the seawater wedge and by numerical integration over the wedge. Mualem and Bear (1974) looked at steady state flow in an aquifer where

a thin semipervious zone divides the aquifer into two layers. Since they used the Dupuit approximation, they allowed only vertical flow in the semipervious zone. Verou (1978) and Hashish et al. (1979) considered a leaky layer at the top of the aquifer and studied the steady state position of the interface for several types of leakage and inland boundary conditions.

Several other authors, also using the sharp interface/hydraulic approach, have examined the effects of pumping and/or recharge wells on coastal seawater intrusion in finite depth aquifers. Strack (1976) found the steady state solution for confined and phreatic aquifers, using (Girinskii) potential flow theory. Kishi and Fukuo (1977) looked at steady flow in confined aquifers, by linearizing the freshwater equations above the interface and using Green's Functions. Kashef (1975, 1976) and Kashef and Smith (1975) used the Ghyben-Herzberg approximation and the principle of superposition to estimate the transient effect of recharge wells on piezometric head and interface position in a confined aquifer.

With the exception of Henry (1964b) all these analytical models take a sharp interface approach. Among these only one, Bear and Dagan (1964b), considered dynamics of the sea water. However, almost all of these models are based on simple assumptions about geometry (infinitely deep or of constant thickness), properties (homogeneous and isotropic), and boundary conditions. If these assumptions were not made it would be difficult, if not impossible, to solve the governing equations analytically. Since in practical problems the aquifer geometry and

parameters vary in space, and natural recharge and pumping vary in both time and space, numerical models are required.

2.2.3 Numerical Models

With the advent of high speed digital computers, numerical models have become an attractive solution method for problems that are too complicated to solve analytically. Table 2.3 presents the various seawater intrusion numerical models found in the literature, classified according to the systematization of Table 2.1. The discussion of these models in the following sections is presented separately for fully mixed models and sharp interface models.

2.2.3.1 Fully Mixed Models

Pinder and Cooper (1970) studied the movement of the transition zone in coastal aquifers by analyzing a vertical cross section of the aquifer using flow and mass transport equations. The structure of these two differential equations is different. The flow equation is a parabolic equation which is accurately approximated by finite difference or finite element methods. The mass transport equation has hyperbolic and parabolic terms. Approximation of the hyperbolic advective terms by these methods is usually not very satisfactory, unless the advective terms are less important than the dispersive ones (see Pinder and Gray, 1977). Examining a vertical cross-sectional view of a homogeneous, isotropic confined coastal aquifer, Pinder and Cooper (1970) solved the steady state flow problem using a finite difference scheme with an iterative alternating directions pro-

Table 2.3

Classification of Numerical Models

TYPE OF MODEL		AUTHOR	SPACE VARIABLES	TYPE OF EQUATION		METHOD OF SOLUTION	FLOW EQUATION
				FLOW	MASS TRANSPORT		
FULLY MIXED		Pinder+Cooper (1970)	x,z	steady*	transient	FD/MC**	hydrodynamic
		Lee+Cheng (1974)	x,z	steady*	steady	FE	hydrodynamic
		Segol et al. (1975)	x,z	steady*	transient	FE	hydrodynamic
		Desai+Contractor (1977)	x,y	steady*	transient	FE	hydrodynamic
		INTERCOMP (1976)	x,y,z	steady*	transient	FD	hydrodynamic
				FRESHWATER FLOW	SEAWATER FLOW		
SHARP INTERFACE	LENS	Kono (1974)	x,z	steady	static	FE	hydrodynamic
		Liu+Liggett (1978)	x,z	steady	static	BIEM	hydrodynamic
		Cheng+Hu (1975)	x,z	steady	steady	FE	hydrodynamic
		Fetter (1972)	x,y	steady	static	FD	hydraulic
		BURGEAP (France)	x,y	transient	static	FD	hydraulic
		Rofail (1977)	x,y	transient	transient	FD	hydraulic
		W.R.C. (England)	x,y	transient	transient	FD	hydraulic
		Pinder+Page (1975)	x,y	transient	transient	FE	hydraulic
	TOE	Shamir+Dagan (1971)	x	transient	transient	FD	hydraulic
	LENS/TOE	Present Model – SWIM	x,y	transient	transient	FE	hydraulic

x,y,z – Cartesian coordinates (z vertical) ; MC – method of characteristics ;

FD – finite differences ; BIEM – boundary integral equation method ;

FE – finite elements .

* – Does not account for time varying boundary conditions; only accounts for time varying flow due to changes in the density of the fluid.

** – FD for flow and dispersion and MC for advection.

cedure, and the transient solute transport problem by the method of characteristics for advection and finite differences for dispersion.

In the present work (SWIM) transient flow is considered when the time derivative is present in the governing equation. Pinder and Cooper (1970) have considered variations in time in the flow equations due only to variations in the density of the fluid at a particular position in space. All of the fully mixed models mentioned below use the same approach. For this reason the flow equation in these models will be considered a "steady state" equation in this review.

Lee and Cheng (1974) solved the steady state version of this problem using the finite element method with linear triangular elements and a Raleigh-Ritz procedure. Segol et al. (1975) analyzed the general problem posed by Pinder and Cooper (1970) using the Galerkin formulation and quadrilateral elements, with linear or quadratic sides. Time integration was handled with an iterative implicit finite difference scheme. This model was applied to the Biscayne aquifer, Florida (Segol and Pinder, 1976). Desai and Contractor (1977) present a similar finite element model for coastal aquifers.

Intercomp (1976) developed a 3-D finite difference transient model for the U.S. Geological Survey, to evaluate the effects of liquid waste disposal in aquifers. This model has also been used to examine seawater intrusion in aquifers. However, the results presented in their report are quite discouraging due to the solution's slow convergence and lengthy computations.

2.2.2.2 Sharp Interface Models

Kono (1974) used the finite element method, with a variational formulation and linear triangular elements, to solve the steady state flow for freshwater above a sharp interface considering the effects of vertical flow. He also examined the problem of interface upconing under a pumping well in the zone above the intruding wedge. Liu and Liggett (1978) used the boundary integral equation method to study the interface in an infinitely deep confined aquifer, again considering the effects of vertical flow. Both Kono (1974) and Liu and Liggett (1978) compared their numerical solutions favorably to analytical solutions of the type presented by Henry (1964a) and Bear and Dagan (1964a). A related problem was addressed by Cheng and Hu (1975). They examined the flow of a density stratified liquid through an embankment by looking at a vertical cross section. They solved simultaneous equations for the freshwater and seawater zones using quadratic quadrilateral finite elements.

The remaining sharp interface models are based on the hydraulic approach and all of them, except the last by Shamir and Dagan (1971), are *lens* type models.

Fetter (1972) developed a simple 2-D steady state lens model using finite differences and the Ghyben-Herzberg approximation. BURGEAP, a private society of consulting engineers in France, has a 2-D flow model of a freshwater lens, called TRABISA, based on finite differences (International Groundwater Information Center, Indianapolis, Indiana). A good model of steady flow conditions, it uses the Ghyben-

Herzberg approximation for tansient flow problems. The Water Resources Center at Medmenham Laboratory in England has a 2-D transient flow finite difference model of a freshwater lens, that solves both freshwater and seawater equations simultaneously (D.B. Oakes, personal communication, 1977). A similar model has been described in a publication by Rofail (1977), although there are some questions about the mathematical formulation (Sá da Costa and Wilson, 1978). Pinder and Page (1976) formulated the 2-D transient lens problem in a Galerkin finite element model using linear triangular elements.

The 1-D transient *toe* problem was modeled by Shamir and Dagan (1971) using the sharp interface/hydraulic approach and finite difference techniques. Particular attention was given to the toe of the interface, which was represented by a moving node; that is, a new finite difference mesh was generated at each time step. This code and some of its results are described in Chapter 5 of this report.

Some summary remarks can be made about the existing numerical models. The fully mixed models all use a hydrodynamic flow equation and a steady state flow solution not allowing for time varying boundary conditions (which are very common in groundwater problems). All but one of the sharp interface models apply to *lens* type situations. Most of the sharp interface models use the hydraulic approach and allow transient flow conditions.

It is evident from Table 2.3 that there is no existing general 2-D

sharp interface/hydraulic numerical code to solve the toe problem for steady or transient conditions. Furthermore, most of the models listed in the table are unavailable or poorly documented. The Sea Water Intrusion Model SWIM has been designed to fill these gaps.

2.3 Related Physical Problems

This section presents a brief discussion of two phase problems which are related to the sharp interface approach to seawater intrusion. These problems can be divided into two categories: porous media problems, such as free surface and petroleum reservoir problems; and other problems, such as Stefan problems. In the next two sections some of these problems are presented following these two categories.

2.3.1 Two Phase Flow and Reservoir Problems

Two phase flow problems involve two liquids, or a liquid and a gas. The reservoir problem encountered in petroleum engineering is a typical example of a two phase flow problem. For seawater intrusion in aquifers one is interested in knowing where the interface separating seawater from freshwater is located. In petroleum engineering the general problem is not to calculate where one fluid ends and where the other begins, but rather to calculate the amount of each fluid found throughout the reservoir, that is, the relative saturation of water to oil. The petroleum engineer has to predict the ratio of water to oil production at a given well. To avoid confusion, the following definitions should be compared: in seawater intrusion an immiscible, sharp or abrupt interface is an interface that separates two fluids, fresh-

water and seawater; in petroleum engineering a "sharp front" is a steep relative saturation zone that separates two areas with different water to oil ratios, for example 10% < and > 10%. A reservoir "sharp front" is equivalent to a very steep seawater-freshwater interface. A seawater intrusion problem can have a toe and the seawater phase can disappear inland of that point; a petroleum reservoir does not have a toe, and both phases coexist everywhere. In an oil reservoir the displacement of the relative saturation front is calculated in the same way as the freshwater/seawater interface for a *lens* situation. The relative ratio of freshwater to seawater saturated thickness, say b^f/b^s, is analogous to the relative water to oil saturation, s^w/s^o.

The petroleum engineering literature contains many contributions to the field of resevoir simulation. Only a few of them will be mentioned here. Peaceman (1977) and Crichlow (1977) review the use of finite difference methods for reservoir simulation. Since both books contain extensive bibliographies no further review of finite difference reservoir models will be given here. The remaining literature concerns the finite element methods.

The finite element method was first introduced to petroleum reservoir simulation by Price et al. (1968). It has since become a very powerful tool used to model saturation front displacements. McMichael and Thomas (1973) employed Galerkin's method to examine three-phase multidimensional (oil-water-gas) compressible flow problems. Solving the resulting system of equations by LSOR, they found that for the same size time step, the CPU time required with this method was

larger than that incurred with finite difference. However, with the finite element methods they also could use larger time steps and did not need to resort to "unnatural artifices to assume a stable, convergent solution". Mercer and Faust (1976) give a good review of the finite element reservoir simulation literature for later work, including the use of various "tricks" such as upstream weighting to improve predictions of sharp front displacements. Other related work is reported by Dalen (1976), Spivak et al. (1977) and Lewis et al. (1978). The major emphasis of this work--the sharp front problem--is of little relevance to seawater intrusion becasue there are no steep immiscible interfaces. The other focus of these papers is the time integration scheme, and this is of interest. Most of these reservoir schemes use finite differences in time, and a Newton-Raphson or Modified Newton-Raphson iterative technique to handle equation non-linearities (see Mercer and Faust, 1976).

Lefebvre du Prey and Weill (1974) treat the oil displacement problem using a moving grid. They allow nodal displacement only in the vertical direction for a linear triangular finite element grid. Three zones are described: water saturated, oil saturated, and water + oil between. The two moving boundaries between the three zones are modeled by moving the grid. The moving boundaries are like the toe of an interface--they describe the limit or extent of one of the two phases. It is quite possible that this model could be used to simulate some of the same 2-D *toe* situations for which SWIM has been designed. It is also obvious that this is an entirely different approach to reservoir modeling form that described above.

2.3.2 Stefan Problems

The Stefan problem, or unsteady heat flow with change of phase, usually involves solid and liquid states with an interface between them. This problem can be found in several industrial applications such as: freezing of food, production or melting of ice, solidification of castings, ablation of missile skins, welding of two materials, and the glass industry. It even occurs when considering the evolution of a star. In these cases the term "interface" is analogous to the term "toe" in seawater intrusion. In numerical simulations a moving grid based on the interface (our "toe") is often used to account for the discontinuity of properties (see Crank, 1975, and Fisher and Medland, 1974). Fisher and Medland (1974) present 2-D finite element moving grid solutions, while Crank (1975) demonstrates some principles of moving finite difference grid solutions. Fisher and Medland also examine a 2-D finite element solution for a fixed grid with a special weighting procedure to account for the interface ("toe") movements. Wellford and Ayer (1977) solved the 1-D Stefan problem with a fixed grid, using special elements with a discontinuous interpolation functions, in the zone crossed by the interface ("toe"). Kushner and Walston (1978) used a 2-D fixed linear triangular finite element grid with a special cubic integration formula to evaluate the properties inside those elements containing the interface.

The Stefan problem is very similar to the immiscible interface/hydraulic approach seawater intrusion problem involving a *toe*. The

toe defines a boundary between the single phase water zone and the two phase freshwater above seawater zone. In transient problems the toe moves; that is, it becomes a moving boundary between the two phases, just as the interface in the Stefan problem becomes a moving boundary. The finite difference or finite element mesh can be regenerated to follow this moving boundary, as is usually done, or special algorithms can be adopted to a fixed grid to indirectly account for it.

2.4 Methodology Selected for SWIM

The review presented in Section 2.2 demonstrates that a gap exists in the state-of-the-art of numerical models of seawater intrusion in aquifers. That is, there is no existing 2-D sharp interface/hydraulic approach numerical code to solve the toe problem for steady or transient conditions. SWIM has been designed to fill this gap, furthermore it also simulates lens type freshwater situations.

SWIM uses the finite element method. Preferred over finite differences, finite elements better represent geometric boundaries, allow a more flexible spatial discretization, permit a better handling of non-homogeneities and anisotropies, and reduce storage requirements for data sets due to the smaller number of node points required. Besides, as discussed below, SWIM uses a toe tracking algorithm based on a fixed mesh and the Gauss quadrature points which are necessary for performing spatial finite element integration over the elements. Mixed isoparametric elements with 4 to 8 nodes are employed for space discretization to provide an optimal level of discretization flexibility, with 4-node linear elements in locations with small gradients and higher order elements elsewhere.

The governing equations are integrated in time using an implicit finite difference scheme. This scheme has no restriction on the time step size. Equation non-linearities, due to water table or interface movements, are handled by a modified Newton-Raphson technique similar to the schemes employed in petroleum reservoir problems (see Section 2.3), and in other non-linear continuum mechanics fields (see, e.g., Desai and Abel, 1972; Zienkiewicz, 1977; Bathe, 1979).

To track the toe movement, accurately with a fixed finite element mesh, a special algorithm was developed. This algorithm uses a non-linear description of hydraulic properties inside those elements containing a toe, and the Gauss quadrature points to indirectly track the toe. It is more sophisticated than somewhat similar ideas described by Wellford and Ayer (1977) for the 1-D Stefan problem, or just a few months ago by Kushner and Walston (1978) for the 2-D Stefan problem. Bathe and Khoshgoftaar (1979) used a closely related technique to model free surface seepage problems without mesh regeneration.

In summary, SWIM is a Galerkin finite element model using mixed isoparametric elements to model seawater intrusion in aquifers. It is based on a sharp interface/hydraulic approach. Both lens and toe situations are modeled, with separate flow equations for freshwater and seawater. The toe movement is indirectly tracked from a fixed mesh by a special algorithm that employs the Gauss quadrature points. An implicit finite difference time integration scheme is used; equation non-linearities are handled by a modified Newton Raphson method. The following chapters describe SWIM in detail.

Chapter 3

MODEL FORMULATION

3.1 Introduction

The general procedure used in this work to solve the problem of seawater intrusion in coastal aquifers is discussed in this chapter. First the governing differential equation and the principal assumptions behind its derivation are introduced. Next the boundary conditions required for solution of the governing equations are discussed. The section that follows presents the derivation of the Galerkin statement for the space integration of the governing equations, and the finite element discretization. Finally, the last part of the chapter deals with the time integration and iterative techniques used to solve the resulting matrix differential equation.

3.2 Governing Equation

The exact mathematical statement for a moving interface between two immiscible liquids is presented in Bear (p. 524-526, 1972). Except for extremely simple problems these equations have no analytical solution and, therefore, require numerical solution.

The derivation of the governing equations used in SWIM is presented in Appendix A. In tensor notation, these differential equations are:

FRESHWATER

$$\frac{\partial}{\partial x_i}\left(K^f_{x_i x_j}\, b^f\, \frac{\partial \phi^f}{\partial x_j}\right) + \frac{K'^f_1}{b'^f_1}\left(\phi'^f_1 - \phi^f\right) + \frac{K'^f_2}{b'^f_2}\left(\phi'^f_2 - \phi^f\right)$$

$$+ q^f + N = \left(S^f + n\frac{\gamma^f}{\Delta\gamma}\right)\frac{\partial \phi^f}{\partial t} - n\frac{\gamma^s}{\Delta\gamma}\frac{\partial \phi^s}{\partial t} \qquad i,j = 1,2 \tag{3.1}$$

SEAWATER

$$\frac{\partial}{\partial x_i}\left(K^s_{x_i x_j}\, b^s\, \frac{\partial \phi^s}{\partial x_j}\right) + \frac{K'^s_1}{b'^s_1}\left(\phi'^s_1 - \phi^s\right) + \frac{K'^s_2}{b'^s_2}\left(\phi'^s_2 - \phi^s\right) + q^s$$

$$= \left(S^s + n\frac{\gamma^s}{\Delta\gamma}\right)\frac{\partial \phi^s}{\partial t} - n\frac{\gamma^f}{\Delta\gamma}\frac{\partial \phi^f}{\partial t} \qquad i,j = 1.2 \tag{3.2}$$

where a number of the terms are illustrated in Figure 3.1. The interface depth, ζ, is defined by the continuity in pressure at both sides of the interface. In terms of piezometric heads, this leads to the expression:

$$\zeta = \frac{\gamma^s}{\Delta\gamma}\,\phi^s - \frac{\gamma^f}{\Delta\gamma}\,\phi^f \tag{3.3}$$

In these equations the superscripts ^{f} and ^{s} represent a freshwater and seawater quantity, respectively, and:

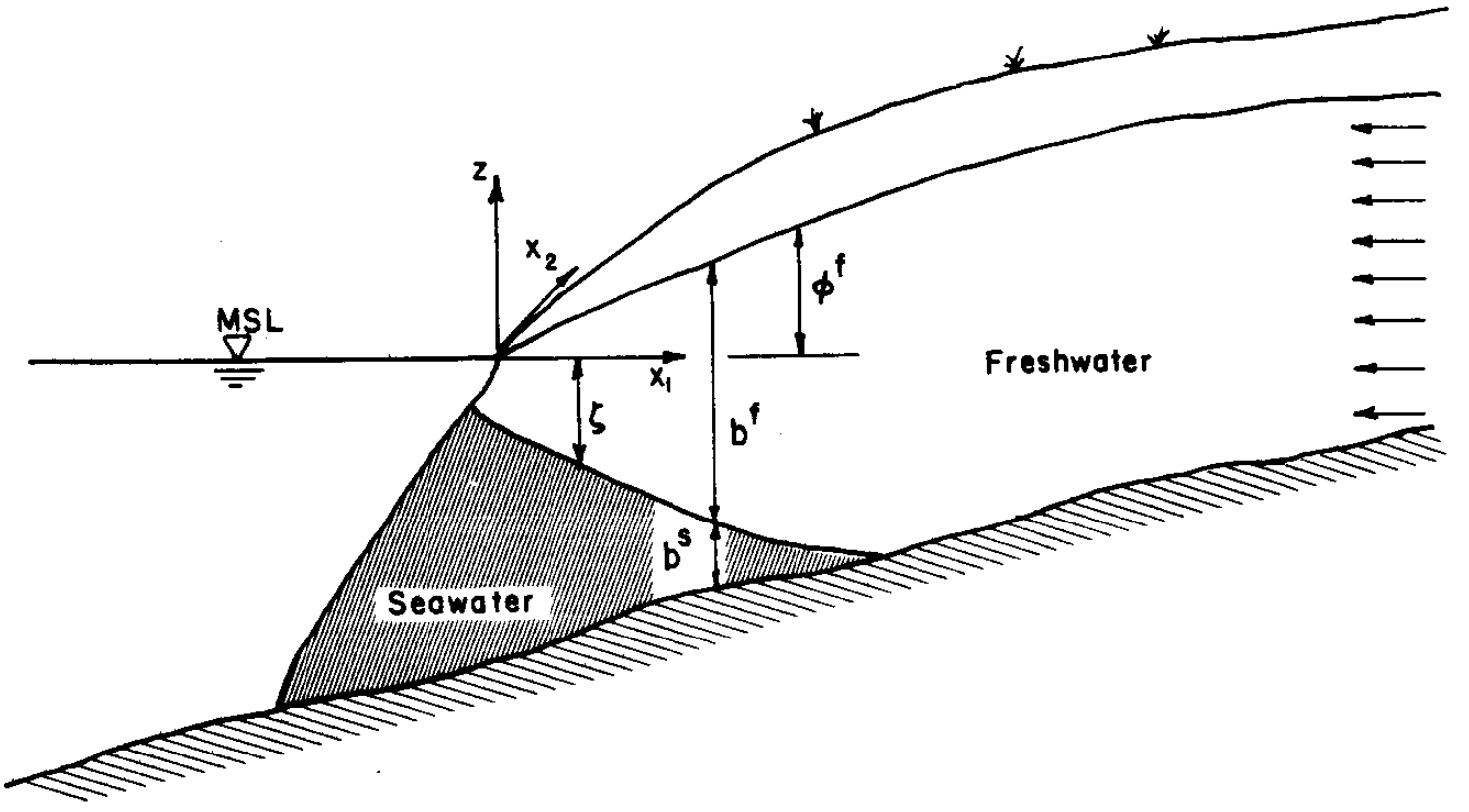


Figure 3.1 - Typical Cross Section of a Coastal Prheatic Aquifer. Definition Sketch.

x_i, x_j - space coordinates $i,j = 1,2$ [L]

$K_{x_i x_j}$ - components of the permeability tensor [L/T]

b - saturated thickness [L], (see Fig. 3.1)

ϕ - piezometric head [L]

K'_m - vertical permeability in the semi-pervious leaky layer (= 0 if nonexistent), m = 1 top layer, m = 2 bottom layer [L/T]

b'_m - thickness of the leaky layer, m = 1,2 [L]

ϕ'_m - piezometric head in a vertically adjacent aquifer, on the other side of the leaky layer, m = 1,2 [L]

q - source/sink term [L/T]

N - natural accretion (= 0 in confined aquifers) [L/T]

S - elastic storativity (confined aquifer) or specific yield (phreatic aquifer) [-]

n - effective porosity [-]

γ - specific weight; $\Delta\gamma = \gamma^s - \gamma^f$ [$M/L^2/T$]

t - time [T]

The major assumptions behind the derivation of these equations are (see Appendix A):

- immisicble interface separating the fresh and seawater phases,
- Darcy's law is applicable,
- vertical variations of storativity, porosity, and horizontal permeability are neglected,
- Dupuit assumption is valid for phreatic aquifers, and essentially horizontal flow in confined aquifers,

- vertical flow is only considered for leaky effects and for accretion,
- constant specific weight for both fresh and seawater.

These two phases are considered homogeneous, isotropic, and they completely fill all voids in the saturated zone of the porous media.

Appendix A also presents the derivation of a matrix algebraic equation, representing Eq. 3.1 and 3.2 in an abbreviated form:

$$\underline{\underline{B}}\ \underline{\underline{T}}\ \underline{\underline{B}}^{T}\underline{\Phi} + \underline{\underline{K}}'_{m}\left(\underline{\Phi}'_{m} - \underline{\Phi}\right) + \underline{Q} = \underline{\underline{S}}\ \frac{\partial \underline{\Phi}}{\partial t} \qquad m=1,2 \tag{3.4}$$

where a matrix quantity is represented by $\underline{\underline{\ }}$ and a vector by $\underline{\ }$. The state vector containing the piezometric head in both freshwater and seawater is:

$$\underline{\Phi} = \begin{bmatrix} \phi^{f} \\ \phi^{s} \end{bmatrix} \tag{3.5}$$

and the other variables are given by:

$$\underline{\underline{B}} = \begin{bmatrix} \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} \end{bmatrix} \tag{3.6}$$

$$\underline{\underline{T}} = \begin{bmatrix} K^f_{x_1x_1} b^f & K^f_{x_1x_2} b^f & 0 & 0 \\ K^f_{x_2x_1} b^f & K^f_{x_2x_2} b^f & 0 & 0 \\ 0 & 0 & K^s_{x_1x_1} b^s & K^s_{x_1x_2} b^s \\ 0 & 0 & K^s_{x_2x_1} b^s & K^s_{x_2x_2} b^s \end{bmatrix} \tag{3.7}$$

$$\underline{\underline{K'}}_m = \begin{bmatrix} K'^f_m / b'^f_m & 0 \\ 0 & K'^s_m / b'^s_m \end{bmatrix} \qquad m = 1,2 \tag{3.8}$$

$$\underline{\underline{\Phi'}}_m = \begin{bmatrix} \phi'^f_m \\ \phi'^s_m \end{bmatrix} \qquad m = 1,2 \tag{3.9}$$

$$\underline{Q} = \begin{bmatrix} q^f + N \\ q^s \end{bmatrix} \tag{3.10}$$

$$\underline{\underline{S}} = \begin{bmatrix} S^f + n\frac{\gamma^f}{\Delta\gamma} & -n\frac{\gamma^s}{\Delta\gamma} \\ -n\frac{\gamma^f}{\Delta\gamma} & S^s + n\frac{\gamma^s}{\Delta\gamma} \end{bmatrix} \tag{3.10}$$

A superscript ^{T} represents a transpose quantity. The matrix $\underline{B}$ is a derivative operator, operating on all terms to its right enclosed within the same parentheses.

3.3 Boundary Conditions

The governing differential equations, Eq. 3.1 and 3.2, are applied to a finite aquifer domian. This domain is defined by an external boundary s_b, which is usually broken into three segments, representing three boundary types.

$$s_b = s_1 + s_2 + s_3 \tag{3.12}$$

For some particular problems one or more of these individual segments may not exist. The three types of boundary conditions are: *specified piezometric head* along boundary segment s_1, also known as a *first type*, Dirichelet, essential or geometric boundary condition; *specified flux* along boundary segment s_2 equivalent to a specification of the first space derivative of piezometric head, also known as a *second type*, Neumann, natural or force boundary condition; and a *mixed boundary condition* along boundary segment s_3, that is defined by specifying a head in an adjacent aquifer or surface water body, which causes a flow through a semipervious layer in or out of the main aquifer; this is also known as *third type* or Cauchy boundary condition. The formulation used in this model incorporates this third type boundary condition in the leakage terms of the governing equations.

In a coastal aquifer the *sea boundary* requires special treatment. In this work the sea boundary condition is treated as a third type boundary condition. That is, it is assumed that the freshwater flows or leaks to the sea through a finite thickness opening along the coastline (see Fig. 3.2). Since the freshwater piezometric level at the

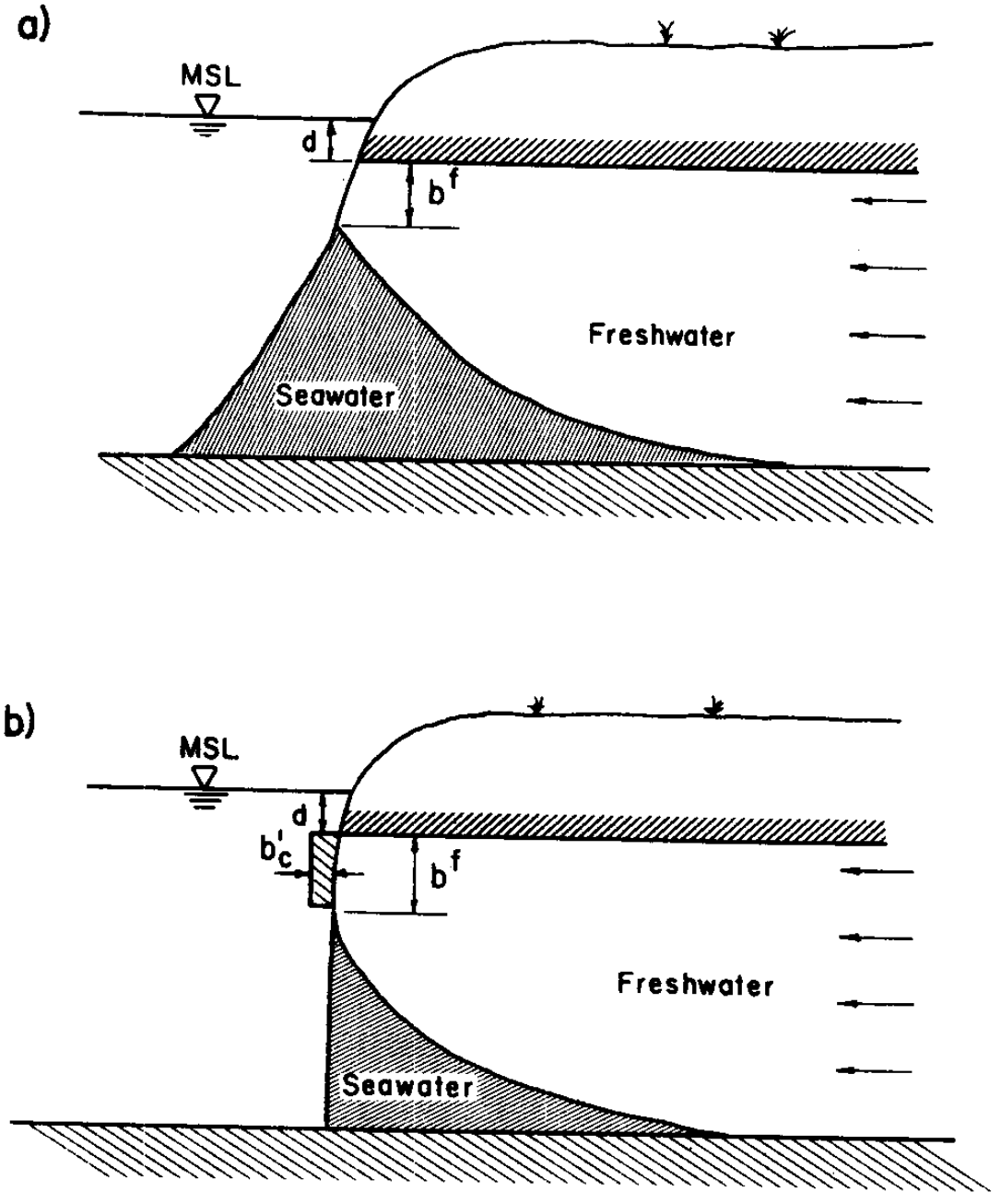


Figure 3.2 - Coastal Boundary Condition: a) Real Situation;
b) SWIM Representation.

sea is zero, all of the remaining freshwater head is lost in this process, which primarily represents vertical flow. The freshwater flow per unit length of coastline toward the sea (out of the aquifer) under these assumptions is:

$$Q_c^f = -\frac{K_c'^f}{b_c'} (\phi_c' - \phi^f)\, b^f \qquad [L^2/T]$$

where the subscript $_c$ stands for *coastal* boundary.

At the coastal boundary the freshwater head is zero, $\phi_c' = 0$, and this equation becomes:

$$Q_c^f = \frac{K_c'^f}{b_c'} b^f \phi^f = K_c' \phi^f \qquad [L^2/T] \tag{3.13}$$

The problem now consists in determining the value of K_c', a function of parameters $K_c'^f$ and b_c', and temporal variable b^f, the opening to the sea ($K_c' = K_c'^f\, b^f/b_c'$).

Bear and Dagan (1964a) developed an exact solution for steady state freshwater flow to the sea for a confined aquifer, accounting for vertical flow, i.e., without taking the Dupuit approximation. They found

$$Q_c^f = K \frac{\Delta\gamma}{\gamma^f} b^f \tag{3.14}$$

Substituting Eq. 3.3 with $\phi^s = 0$ in Eq. 3.14 gives:

$$Q_c^f = K\phi^f - K \frac{\Delta\gamma}{\gamma^f} d \qquad (3.15)$$

where K is the permeability of the aquifer, and d the depth below mean sea level of the top of the confining layer as shown in Fig. 3.2, d = 0 for phreatic aquifers. Ignoring the last term in Eq. 3.15 leads to an expression similar to Eq. 3.13:

$$Q_c^f = K\phi^f \qquad [L^2/T]$$

Therefore, in this case, $K_c' = K$, and is a constant. Subsequent analysis in this work assumed that K_c' is constant, and it is often set equal to K. With this simplified assumption the sea boundary condition is treated as a linear boundary in the parameter K_c' rather than as non-linear. Rumer and Harleman (1963) and Pinder and Page (1976) used similar assumptions to obtain an opening to the sea.

Some authors also regard the initial conditions as a boundary condition. They treat these conditions as a first type boundary condition in time. In steady state problems initial conditions are usually not necessary, but due to the nonlinearities of the problem under discussion, an initial guess is required. This initial *guess* is called an "initial condition" hereafter, and involves the knowledge of an initial position or guess of the interface and the corresponding freshwater piezometric head.

3.4 Galerkin Approximation of the Governing Equations

Putting all the known quantities on the right hand side, the governing equation, Eq. 3.4, can be written as:

$$\underline{\underline{S}} \frac{\partial \underline{\Phi}}{\partial t} - \underline{\underline{B}}\,\underline{\underline{T}}\,\underline{\underline{B}}^T \underline{\Phi} + \underline{\underline{K}}'_m \underline{\Phi} = \underline{Q} + \underline{\underline{K}}'_m \underline{\Phi}'_m \qquad m=1,2 \qquad (3.16)$$

or in a simplified way as

$$L(\underline{\Phi}) = \underline{q} \qquad (3.17)$$

where $L(\ )$ is an operator and $\underline{q}$ represents the independent variables.

The continuous state vector $\underline{\Phi}$ is approximated by a *trial function* $\underline{\hat{\Phi}}$:

$$\underline{\Phi} \simeq \underline{\hat{\Phi}} = \underline{\underline{H}}\,\underline{X} \qquad (3.18)$$

where

$$\underline{\hat{\Phi}} = \begin{bmatrix} \hat{\phi}^f \\ \\ \hat{\phi}^s \end{bmatrix} \qquad (3.19)$$

and

$$\phi^f \simeq \hat{\phi}^f = \sum_{i=1}^{n} h_i \phi_i^f \qquad (3.20)$$

$$\phi^s \simeq \hat{\phi}^s = \sum_{i=1}^{n} h_i \phi_i^s \qquad (3.21)$$

The matrix, $\underline{\underline{H}} = \underline{\underline{H}}(x_1,x_2)$ is called an interpolation or shape function, and is a function of space but not time. Its definition is:

$$\underline{\underline{H}} = \begin{bmatrix} h_1 & 0 & h_2 & 0 \ldots & h_n & 0 \\ 0 & h_1 & 0 & h_2 \ldots & 0 & h_n \end{bmatrix} \tag{3.22}$$

The vector $\underline{X} = \underline{X}(t)$ represents the values of the piezometric heads for both freshwater and seawater phases at n discrete points called node points; they are only a function of time. $\underline{X}$ is a symbolic representation of the vectors of piezometric head, $\underline{\phi}^f$ and $\underline{\phi}^s$:

$$\underline{X} = \begin{bmatrix} \phi_1^f \\ \phi_1^s \\ \phi_2^f \\ \phi_2^s \\ \vdots \\ \phi_n^f \\ \phi_n^s \end{bmatrix} \tag{3.23}$$

where ϕ_i^f and ϕ_i^s are freshwater and seawater head, respectively, at node point i. The particular construction of $\underline{X}$, with <u>alternating</u> values of freshwater and seawater piezometric heads, and the corresponding construction of $\underline{\underline{H}}$, is due to the fact that they minimize the bandwidth of the matrices involved. Given the definition of $\underline{\underline{H}}$ and $\underline{X}$, the following rules for derivatives hold:

$$\underline{\underline{B}}^T (\underline{\underline{H}}\ \underline{X}) = (\underline{\underline{B}}^T\ \underline{\underline{H}})\ \underline{X} \tag{3.24}$$

$$\frac{\partial}{\partial t} (\underline{\underline{H}}\ \underline{X}) = \underline{\underline{H}}\ \frac{d\underline{X}}{dt} \tag{3.25}$$

Introducing the trial function $\underline{\hat{\Phi}}$ in Eq. 3.17 in place of state vector $\underline{\Phi}$ results in a small error ε:

$$\underline{\varepsilon} = L(\underline{\hat{\Phi}}) - \underline{q} \neq \underline{0} \tag{3.26}$$

The weighted residuals approach seeks to determine the nodal values of piezometric heads, $\underline{X}$, by setting a weighted integral of Eq. 3.26 over the entire domain area, A, equal to zero. That is:

$$\int_A w_i\, \underline{\varepsilon}\, dA = \int_A w_i [L(\underline{\hat{\Phi}}) - \underline{q}]\, dA = \underline{0} \qquad i=1,2,\ldots,n \tag{3.27}$$

where w_i are arbitrary weighting functions and $\underline{0}$ is the two-dimensional null vector. In order to use a matrix notation let us introduce the matrix $\underline{\underline{W}}$

$$\underline{\underline{W}} = \begin{bmatrix} w_1 & 0 & w_2 & 0 \ldots & w_n & 0 \\ 0 & w_1 & 0 & w_2 \ldots & 0 & w_n \end{bmatrix} \tag{3.28}$$

so that Eq. 3.27 can be rewritten as:

$$\int_A \underline{\underline{W}}^T \underline{\varepsilon}\, dA = \int_A \underline{\underline{W}}^T [L(\hat{\underline{\phi}}) - \underline{q}]\, dA = \underline{0} \tag{3.29}$$

Substituting Eq. 3.16 and 3.18 in Eq. 3.29 leads to:

$$\int_A \underline{\underline{W}}^T \left[\underline{\underline{S}}\ \underline{\underline{H}}\ \frac{d\underline{\underline{X}}}{dt} - \underline{Q} - \underline{\underline{K}}'_m\ \underline{\underline{H}}\ (\underline{X}'_m - \underline{X}) - \underline{\underline{B}}\ \underline{\underline{T}}\ \underline{\underline{B}}^T\ \underline{\underline{H}}\ \underline{X} \right] dA = 0 \qquad m = 1,2 \tag{3.30}$$

where

$$\underline{X}'_m = \begin{bmatrix} \phi_1'^f \\ \phi_1'^s \\ \phi_2'^f \\ \phi_2'^s \\ \vdots \\ \phi_n'^f \\ \phi_n'^s \end{bmatrix}_m$$

This equation has a term with second order derivatives, that can be reduced in order by using Green's theorem. Looking only at this term:

$$\int_A \underline{\underline{W}}^T (\underline{\underline{B}}\ \underline{\underline{T}}\ \underline{\underline{B}}^T\ \underline{\underline{H}}\ \underline{X})\, dA = \int_{s_b} \underline{\underline{W}}^T (\underline{\ell}\ \underline{\underline{T}}\ \underline{\underline{B}}^T\ \underline{\underline{H}}\ \underline{X})\, ds - \int_A (\underline{\underline{B}}^T\ \underline{\underline{W}})^T (\underline{\underline{T}}\ \underline{\underline{B}}^T\ \underline{\underline{H}}\ \underline{X})\, dA \tag{3.31}$$

where s_t is the exterior geometric boundary,

$$\underline{\ell} = \begin{bmatrix} \ell_{x_1} & \ell_{x_2} & 0 & 0 \\ 0 & 0 & \ell_{x_1} & \ell_{x_2} \end{bmatrix} \tag{3.32}$$

and ℓ_{x_j}, $j = 1,2$, are the unit normal components in the x_j direction.

If the weighting functions w_i are restricted to those which vanish along the first type boundary condition segments, s_1, then the limit of integration on the first integral on the right-hand side of Eq. 3.31 can be changed to s_2 by using Eq. 3.12 and because $s_3 = 0$, since the third type boundary conditions are incorporated in the equation. This integral represents the second type boundary conditions, the flow across the boundary segment s_2. Using Eqs. 3.6 through 3.11 and Eq. 3.30, and a similar procedure to that outlined in Appendix A (Section A.3) gives

$$\underline{\ell}\ \underline{T}\ \underline{B}^T\ \underline{H}\ \underline{X} = \underline{\ell}\ \underline{T}\ \underline{B}^T\ \underline{\phi} = \begin{bmatrix} \ell_{x_i} K^f_{x_i x_j} b^f \dfrac{\partial \phi^f}{\partial x_j} \\ \ell_{x_i} K^s_{x_i x_j} b^s \dfrac{\partial \phi^s}{\partial x_j} \end{bmatrix} = \begin{bmatrix} -q^{*f} \\ -q^{*s} \end{bmatrix} = -\underline{q}^* \tag{3.33}$$

where the superscript * represents a specified quantity, in this case specified fluxes, q^{*f} and q^{*s}, for both freshwater and seawater respectively. The flow is positive into the aquifer. Substituting Eq. 3.33 in Eq. 3.31, changes the latter to

$$\int_A \underline{W}^T(\underline{B}\ \underline{T}\ \underline{B}^T\underline{HX})\,dA = -\int_{s_2} \underline{W}^T\underline{q}^*\,ds - \int_A (\underline{B}^T\underline{W})^T(\underline{T}\ \underline{B}^T\underline{HX})\,dA \tag{3.34}$$

Applying this result to Eq. 3.30 leads to

$$\int_A \left\{ \underline{\underline{W}}^T \left[\underline{\underline{S}}\ \underline{\underline{H}}\ \frac{d\underline{X}}{dt} - \underline{Q} - \underline{\underline{K}}'_m\ \underline{\underline{H}}(\underline{X}'_m - \underline{X}) \right] + (\underline{\underline{B}}^T \underline{\underline{W}})^T (\underline{\underline{T}}\ \underline{\underline{B}}^T \underline{\underline{H}}\underline{X}) \right\} dA +$$

$$+ \int_{s_2} \underline{\underline{W}}^T \underline{q}^* ds = \underline{0} \qquad m=1,2 \tag{3.35}$$

which is the new form of the governing equations, containing only first order derivatives and incorporating the second type boundary conditions. In Galerkin's method, the weighting functions are taken as the interpolation functions, that is:

$$\underline{\underline{W}} = \underline{\underline{H}} \tag{3.36}$$

or for node i, $w_i = h_i$. Substituting $\underline{\underline{H}}$ for $\underline{\underline{W}}$ in Eq. 3.35, the following expression is obtained after rearranging the terms:

$$\left\{ \int_A \underline{\underline{H}}^T \underline{\underline{S}}\ \underline{\underline{H}}\ dA \right\} \frac{d\underline{X}}{dt} - \int_A \underline{\underline{H}}^T \underline{Q}\ dA - \left\{ \int_A \underline{\underline{H}}^T \underline{\underline{K}}'_m\ dA \right\} (\underline{X}'_m - \underline{X})$$

$$+ \left\{ \int_A (\underline{\underline{B}}^T \underline{\underline{H}})^T (\underline{\underline{T}}\ \underline{\underline{B}}^T \underline{\underline{H}})\ dA \right\} \underline{X} + \int_{s_2} \underline{\underline{H}}_s^T \underline{q}^*\ ds = 0 \qquad m=1,2 \tag{3.37}$$

$\underline{\underline{H}}_s$ are the interpolation functions along the boundary segments. Note that all space derivatives are in terms of derivatives of the interpolation functions $\underline{\underline{H}}$. The following simplified notation is useful:

$$\underline{\underline{\mathbf{B}}} = \underline{\underline{B}}^T \underline{\underline{H}} \tag{3.38}$$

Therefore Eq. 3.37 can be written as:

$$\left\{\int_A \underline{\underline{H}}^T \underline{\underline{S}}\ \underline{\underline{H}}\ dA\right\}\frac{d\underline{X}}{dt} - \int_A \underline{\underline{H}}^T \underline{Q}\ dA - \left\{\int_A \underline{\underline{H}}^T \underline{\underline{K}}'_m\ dA\right\}(\underline{X}'_m - \underline{X})$$

$$+\left\{\int_A \underline{\underline{B}}^T \underline{\underline{T}}\ \underline{\underline{B}}\ dA\right\}\underline{X} + \int_{s_2} \underline{\underline{H}}_s^T\, \underline{q}^*\ ds = 0 \qquad m=1,2 \qquad (3.39)$$

The analytical evaluation of these integrals is usually impossible for distorted elements, and a numerical integration is required. This integration is regarded as a part of the isoparametric element matrix computations. In SWIM Gaussian quadrature is used. $\underline{\underline{T}}$ is treated as a functional coefficient, see Mercer and Faust (1976), in that b^f and b^s vary over each element.

The following definitions help to simplify Eq. 3.39:

$$\underline{\underline{C}} = \int_A \underline{\underline{H}}^T \underline{\underline{S}}\ \underline{\underline{H}}\ dA \qquad (3.40a)$$

$$\underline{Q} = \int_A \underline{\underline{H}}^T \underline{Q}\ dA - \int_{s_2} \underline{\underline{H}}^T \underline{q}^* ds \qquad (3.40b)$$

$$\underline{\underline{L}}_m = \int_A \underline{\underline{H}}^T \underline{\underline{K}}'_m\ dA \qquad m=1,2 \qquad (3.40c)$$

$$\underline{\underline{T}} = \int_A \underline{\underline{B}}^T \underline{\underline{T}}\ \underline{\underline{B}}\ dA \qquad (3.40d)$$

where $\underline{\underline{C}}$ is a storage matrix, $\underline{Q}$ the total flow vector, $\underline{\underline{L}}_m$ the third type boundary matrices*, and $\underline{\underline{T}}$ the conductivity matrix. Writing Eq. 3.39

---

*The matrices $\underline{\underline{L}}$ as described by Eq. 3.40b represent the leakage terms consistently, if they are specified over an element area or along an element side. However, the leakage can be represented in SWIM in a lumped fashion by specifying leakage parameters at the nodes.

using these definitions and transferring to the right-hand side all known quantities gives:

$$\underline{\underline{C}}\,\frac{d\underline{X}}{dt} + \underline{\underline{T}}\,\underline{X} + \underline{\underline{L}}_m\,\underline{X} = \underline{Q} + \underline{\underline{L}}_m\,\underline{X}' \qquad m=1,2 \tag{3.41}$$

or in a simpler form

$$\underline{\underline{C}}\,\frac{d\underline{X}}{dt} + \underline{\underline{K}}\,\underline{X} = \underline{F} \tag{3.42}$$

where the total conductivity matrix is given by

$$\underline{\underline{K}} = \underline{\underline{T}} + \underline{\underline{L}}_1 + \underline{\underline{L}}_2 \tag{3.43a}$$

and the force vector by

$$\underline{F} = \underline{Q} + \underline{\underline{L}}_1\underline{X}'_1 + \underline{\underline{L}}_2\underline{X}'_2 \tag{3.43b}$$

Eq. 3.42, a matrix differential equation, represents the final form of the governing equations. Time integration of this equation is discussed in the following sections.

In the models using the finite element method the matrices defined in Eqs. 3.40 are evaluated at the element level, that is the limit of integration is A^e, the area of each individual element. The global matrices are obtained by adding together the element contributions at each node, from all the elements that share that node. The global matrices are banded and have a dimension of (2n x 2n), where n is the total number of nodes used to discretize the domain. The size of the bandwidth depends on the order in which the nodes are numbered.

3.5 Finite Element Discretization

A subject of major importance in finite element analysis is the discretization of the domain, involving decisions about the number, size and type of elements to use. A balance must be struck between many small elements, giving good accuracy, and a few large elements, reducing computational effort. A general feeling of how the aquifer is going to behave is always helpful, because the element size should be reduced in areas with large piezometric gradients, and increased in regions with relatively constant piezometric heads. In numerical modeling, a judicious discretization of a domain comes with experience.

The most general element used in SWIM is the eight node isoparametric element represented in Fig. 3.3a. This element is shown in terms of the local coordinates (r,s) in Fig. 3.3b. The interpolation functions, Fig. 3.3c , for this element are:

$$h_1 = \frac{1}{4}(1+r)(1+s) - \frac{h_5}{2} - \frac{h_8}{2} = \frac{1}{4}(1+r)(1+s)(r+s-1)$$

$$h_2 = \frac{1}{4}(1-r)(1+s) - \frac{h_5}{2} - \frac{h_6}{2} = \frac{1}{4}(1-r)(1+s)(s-r-1)$$

$$h_3 = \frac{1}{4}(1-r)(1-s) - \frac{h_6}{2} - \frac{h_7}{2} = \frac{1}{4}(1-r)(1-s)(-s-r-1)$$

$$h_4 = \frac{1}{4}(1+r)(1-s) - \frac{h_7}{2} - \frac{h_8}{2} = \frac{1}{4}(1+r)(1-s)(r-s-1)$$

$$h_5 = \frac{1}{2}(1-r^2)(1+s)$$

$$h_6 = \frac{1}{2}(1-s^2)(1-r)$$

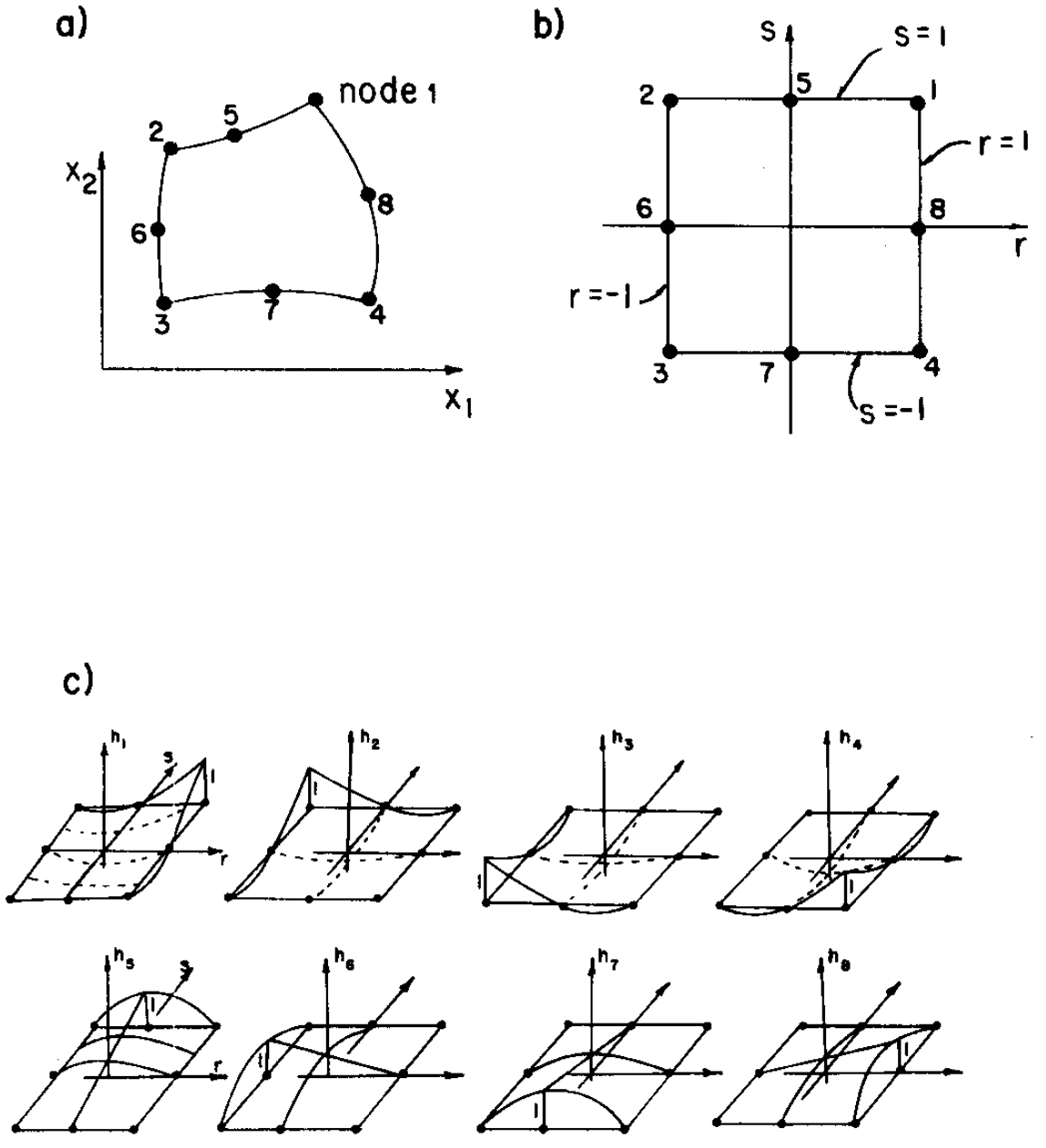


Figure 3.3 - General 8-node Element in a) Global Coordinates; b) Local Coordinates; c) with Associated Interpolation Functions.

$$h_7 = \frac{1}{2}(1-r^2)(1-s)$$

$$h_8 = \frac{1}{2}(1-s^2)(1+r)$$

The transformation between the local and global coordinates is achieved using the Jacobian matrix $\underline{\underline{J}}$: defined as:

$$\underline{\underline{J}} = \begin{bmatrix} \frac{\delta x_1}{\delta r} & \frac{\delta x_2}{\delta r} \\ \frac{\delta x_1}{\delta s} & \frac{\delta x_2}{\delta s} \end{bmatrix} \tag{3.44}$$

The general eight-node element can degenerate in other types of elements having from 4 to 8 nodes, see Fig. 3.4. Triangular elements are obtained by collapsing the corner node numbers 1 and 4; thus the three node triangular element actually uses 4 nodes in SWIM.

The theory of finite elements is fully documented in the literature (Zienkiewicz, 1977; Bathe and Wilson, 1976; Huebner, 1975; Desai and Abel, 1972), with some having special emphasis on the application to groundwater (Connor and Brebbia, 1976; Pinder and Gray, 1977).

3.6 Time Integration

The finite difference method is used to evaluate the time derivative of the matrix differential equation, Eq. 3.42. In groundwater hydrology the more common techniques for approximating this derivative are:

Explicit: $$\left.\frac{d\underline{X}}{dt}\right|_t = \frac{\underline{X}_{t+\Delta t} - \underline{X}_t}{\Delta t} \tag{3.45}$$

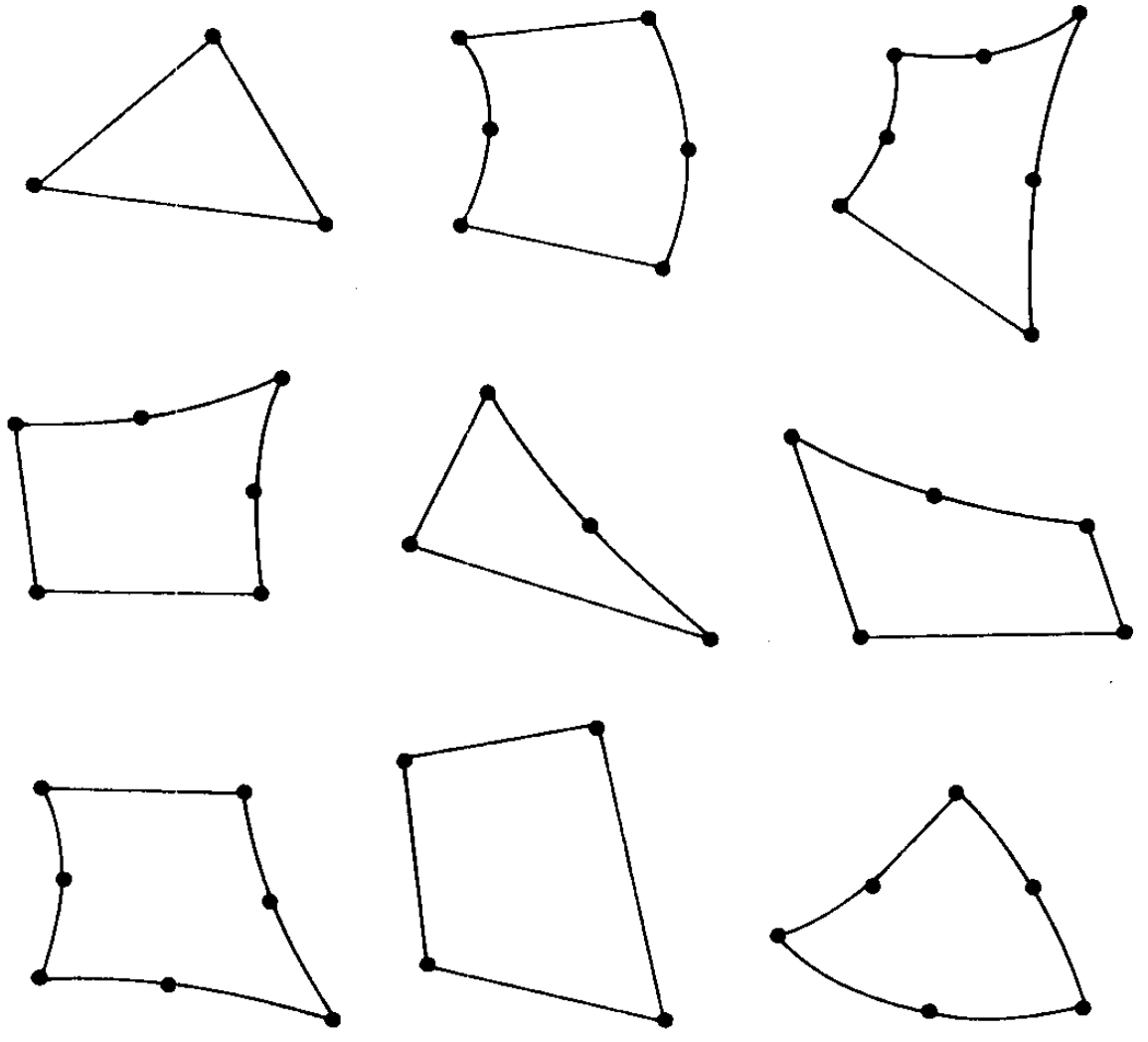


Figure 3.4 - 2-D Elements Derived from the General 8-node Element in Fig. 3.3

Implicit: $$\left.\frac{d\underline{X}}{dt}\right|_{t+\Delta t} = \frac{\underline{X}_{t+\Delta t} - \underline{X}_t}{\Delta t} \qquad (3.46)$$

Trapezoidal rule or

Crank-Nicholson: $$\left.\frac{d\underline{X}}{dt}\right|_{t} = \frac{\underline{X}_{t+\Delta t} - \underline{X}_{t-\Delta t}}{2\Delta t} \qquad (3.47)$$

The subscripts on these equations refer to the time at which the subscripted quantities are evaluated.

The *explicit method* has the advantage of expressing $\underline{X}_{t+\Delta t}$ only in terms of known quantities. Using Eq. 3.45 in Eq. 3.42, this leads to:

$$\underline{\underline{C}}\ \underline{X}_{t+\Delta t} = -(\Delta t\ \underline{\underline{K}}_t - \underline{\underline{C}})\underline{X}_t + \Delta t\ \underline{F}_t \qquad (3.48)$$

Matrix $\underline{\underline{K}}$ is subscripted in time because of the non-linearity of the solution. It must be continually updated during the solution; in this case, using the last calculated values of the piezometric heads and saturated thickness. Matrix $\underline{\underline{C}}$ is constant in time, unless the aquifer undergoes a change in status from confined to phreatic or vice-versa*. In many problems the matrix $\underline{\underline{C}}$ can be evaluated in a *lumped or diagonalized* form. When this is possible the explicit method has an advantage: it does not require the assembly and reduction of the coefficient matrices in order to solve for the head. On the other hand, this method constrains the size of the time step, due to an intrinsic stability criterion. To determine the maximum time step exactly requires the evaluation of the eigenvalues of matrices $\underline{\underline{K}}$ and $\underline{\underline{C}}$, which are time

*Temporal variations in the storage part of $\underline{\underline{C}}$ are negligible compared to porosity storage yielded by interface movement.

and problem dependent. On a similar type of model and assuming optimistic values Shamir and Dagan (1971) found a maximum time step of the order of 8 days, too small to be practical for most applications considered. Due to the particular form of the storage matrix encountered in the seawater intrusion problem (see matrix $\underline{\underline{S}}$, Eq. 3.11, and note that it is not a diagonal matrix), a lumped storage matrix $\underline{\underline{C}}$ will not be diagonal, but tri-diagonal. Therefore, the primary advantage of the explicit formulation of not requiring the assembly of the matrices is lost, although a tri-diagonal algorithm, say the Thomas algorithm, could be used. Because of this and the small time step allowed, the explicit formulation was abandoned.

From the other two options, the implicit method was selected for the existing time integration scheme. Using the implicit scheme Eq. 3.42 can be written:

$$\underline{\underline{C}} \frac{\underline{X}_{t+\Delta t} - \underline{X}_t}{\Delta t} + \underline{\underline{K}}_{t+\Delta t} \underline{X}_{t+\Delta t} = \underline{F}_{t+\Delta t} \tag{3.49}$$

where $\underline{\underline{C}}$ is assumed constant in time. Transferring all known quantities to the right-hand side the following matrix equation is obtained:

$$\left(\frac{\underline{\underline{C}}}{\Delta t} + \underline{\underline{K}}_{t+\Delta t}\right) \underline{X}_{t+\Delta t} = \frac{\underline{\underline{C}}}{\Delta t} \underline{X}_t + \underline{F}_{t+\Delta t} \tag{3.50}$$

Since this is a nonlinear system of equations, the solution is obtained by iteration. The iterative procedure used is discussed in the next section. First, however, note by examining Eqs. 3.40 and 3.43 as well

as Eqs. 3.6 through 3.9 that all matrices involved in the formation of matrix $\underline{\underline{K}}$ are symmetric matrices of the same order as $\underline{\underline{K}}$, and therefore $\underline{\underline{K}}$ is also a symmetric matrix. However, matrix $\underline{\underline{C}}$ is not symmetric, because the original elemental storage matrix, Eq. 3.11, is not symmetric, preventing the use of algorithms typical of symmetric systems in the solution of the system of equations (3.40).

Symmetric matrix algorithms are faster than those for non-symmetric matrices, and only require storage of the upper triangular part of the matrix. Looking more carefully at matrix $\underline{\underline{S}}$ as defined earlier;

$$\underline{\underline{S}} = \begin{bmatrix} S^f + \frac{n\gamma^f}{\Delta\gamma} & -n\frac{\gamma^s}{\Delta\gamma} \\ -n\frac{\gamma^f}{\Delta\gamma} & S^s + n\frac{\gamma^s}{\Delta\gamma} \end{bmatrix} \tag{3.11}$$

one can see that this matrix is only slightly non-symmetric, that is the difference between the non-diagonal terms for the case of freshwater and seawater is of the order of 3% ($\gamma^s/\gamma^f \simeq 1.03$). One way of avoiding the lack of symmetry is to consider two storage matrices: one as defined by Eq. 3.40a, and the other using a symmetric matrix $\underline{\underline{S}}^*$ defined as

$$\underline{\underline{S}}^* = \begin{bmatrix} S^f + n\frac{\gamma^f}{\Delta\gamma} & -n\frac{\gamma^s}{\Delta\gamma} \\ -n\frac{\gamma^s}{\Delta\gamma} & S^s + n\frac{\gamma^s}{\Delta\gamma} \end{bmatrix} \tag{3.51}$$

to generate a symmetric storage matrix defined by:

$$\underline{\underline{C}}^* = \int_{A^e} \underline{\underline{H}}^T \, \underline{\underline{S}}^* \, \underline{\underline{H}} \, dA \tag{3.52}$$

Introducing this matrix in the left-hand side of Eq. 3.50 gives

$$\left(\frac{\underline{\underline{C}}^*}{\Delta t} + \underline{\underline{K}}_{t+\Delta t}\right) \underline{X}_{t+\Delta t} = \left(\underline{\underline{C}}\,\frac{\underline{X}_t}{\Delta t} + \underline{F}_{t+\Delta t}\right) \tag{3.53}$$

where now the matrix inside the parentheses on the left-hand side is symmetric*. Since Eq. 3.53 is solved iteratively this approximation will eventually increase slightly the number of iterations, but the savings resulting from it are much greater.

Due to the particular form of the matrix $\underline{\underline{C}}$, only part of it need be stored. One can consider $\underline{\underline{S}}$ as being a summation of the following matrices:

$$\begin{aligned}
\underline{\underline{S}} &= \begin{bmatrix} S^f + n\,\frac{\gamma^f}{\Delta\gamma} & -n\,\frac{\gamma^s}{\Delta\gamma} \\ -n\,\frac{\gamma^f}{\Delta\gamma} & S^s + n\,\frac{\gamma^s}{\Delta\gamma} \end{bmatrix} \\
&= \begin{bmatrix} S^f + n\,\frac{\gamma^f}{\Delta\gamma} & 0 \\ 0 & S^s + n\,\frac{\gamma^s}{\Delta\gamma} \end{bmatrix} \\
&+ \begin{bmatrix} 0 & -n\,\frac{\gamma^s}{\Delta\gamma} \\ 0 & 0 \end{bmatrix} \\
&+ \begin{bmatrix} 0 & 0 \\ -n\,\frac{\gamma^f}{\Delta\gamma} & 0 \end{bmatrix}
\end{aligned} \tag{3.54}$$

*In Appendix B the positive definiteness of matrix $\left(\frac{\underline{\underline{C}}^*}{\Delta t} + \underline{\underline{K}}_{t+\Delta t}\right)$ is verified.

or $\underline{\underline{S}} = \underline{\underline{S}}_D + \underline{\underline{S}}_U + \underline{\underline{S}}_L$

and the true storage matrix can be represented by:

$$\underline{\underline{C}} = \int_{A^e} \underline{\underline{H}}^T \underline{\underline{S}}_D \underline{\underline{H}} \, dA + \int_{A^e} \underline{\underline{H}}^T \underline{\underline{S}}_U \underline{\underline{H}} \, dA + \int_{A^e} \underline{\underline{H}}^T \underline{\underline{S}}_L \underline{\underline{H}} \, dA$$

$$= \underline{\underline{C}}_D + \underline{\underline{C}}_U + \underline{\underline{C}}_L \qquad (3.55)$$

Where the first term is a symmetric matrix, the second term is a matrix symmetric in relation to a diagonal that is displaced one position to the right of the true diagonal, and the third matrix is a symmetric matrix in relation to a diagonal that is displaced one position to the left of the true diagonal. Due to this property only the upper triangular part of these three matrices, $\underline{\underline{C}}_D$, $\underline{\underline{C}}_U$, $\underline{\underline{C}}_L$, needs to be stored.

Another aspect of the storage matrix is that it can be calculated in a lumped way, that is as a tridiagonal matrix instead of as the full banded matrix of the consistent case. However, it is advisable to use the consistent storage matrix to obtain accurate results for seawater intrusion problems, especially with higher order elements, because the savings obtained by using a lumped storage matrix are marginal and the accuracy less satisfactory. Later, comparisons of solutions obtained using both methods are presented.

3.7 Solution of the Non-linear System of Equations

The governing equation has "two non-linearities": one resulting

from the moving interface, the other associated with the water table of a phreatic aquifer. These non-linearities require that an iterative procedure be used to solve the system of equations. Fully iterative methods, like Gauss-Siedel, SIP, LSOR, and ADI, so common in finite difference models are seldom considered in finite elements. The recent improvement of high speed computers has made direct techniques more attractive. Desai and Abel (1972) mention three basic direct techniques for solving non-linear problems by the finite element method: incremental or stepwise procedures, iterative or Newton methods, and step-iterative or mixed procedures. These methods converge in few iterations, usually less than 3-6, and according to Bathe and Wilson (1976) they are almost always more effective.

In SWIM a modified Newton method is used. To better understand the details of this method let's consider as an example a steady state problem for which Eq. 3.42 reduces to:

$$\underline{\underline{K}}\,\underline{X} = \underline{F} \tag{3.56}$$

To solve this problem using an *iterative method* requires knowledge of the applied flow $\underline{F}'$ and an initial guess for the state vector, $\underline{X}^{(0)}$. With this value of $\underline{X}^{(0)}$, a matrix $\underline{\underline{K}}^{(0)}$ is calculated. In the next step an out of balance flow, $\underline{F}_o^{(1)}$ is determined using:

$$\underline{F}_o^{(1)} = \underline{F}' - \underline{\underline{K}}^{(0)}\underline{X}^{(0)} = \underline{F}' - \underline{F}_e^{(0)} \tag{3.57}$$

where $\underline{F}_e^{(0)}$ is the flow equilibrated by the initial guess $\underline{X}^{(0)}$. The

superscripts in this equation represent the iteration number. The out of balance flow is usually different from zero, because the initial guess $\underline{X}^{(0)}$ is not often an exact solution for Eq. 3.56. This out of balance flow is now equilibrated iteratively. The initial guess $\underline{X}^{(0)}$ is modified by solving the following system of equations for $\underline{\Delta X}^{(1)}$:

$$\underline{\underline{K}}^{(0)} \; \underline{\Delta X}^{(1)} = \underline{F}_o^{(1)}$$

where $\underline{\Delta X}^{(1)}$ is the increment on the piezometric head, $\underline{\Delta X}^{(1)} = \underline{X}^{(1)} - \underline{X}^{(0)}$. The matrix $\underline{\underline{K}}$ is evaluated in the previous iteration. That is, $\underline{\underline{K}}^{(0)}$ is the "slope" of the surface $[\underline{F},\underline{X}]$ at the point $[\underline{F}^{(0)}, \underline{X}^{(0)}]$, as illustrated in Fig. 3.5a, for the one-dimensional case. The piezometric head at the end of this iteration is $\underline{X}^{(1)} = \underline{X}^{(0)} + \underline{\Delta X}^{(1)}$. With this new value of $\underline{X}$, the matrix $\underline{\underline{K}}^{(1)} = \underline{\underline{K}}(\underline{X}^{(1)})$ is evaluated and a new out of balance flow is computed using:

$$\underline{F}_o^{(2)} = \underline{F}' - \underline{\underline{K}}^{(1)} \underline{X}^{(1)} = \underline{F}' - \underline{\underline{K}}^{(1)} \underline{X}^{(0)} - \underline{\underline{K}}^{(1)} \; \underline{\Delta X}^{(1)} = \underline{F}' - \underline{F}_e^{(1)}$$

Then $\underline{\Delta X}^{(2)}$ is calculated from $\underline{\underline{K}}^{(1)} \; \underline{\Delta X}^{(2)} = \underline{F}_o^{(2)}$. This procedure is repeated until the increments of the piezometric head, $\underline{\Delta X}^{(i)}$, or the out of balance flow, $\underline{F}_o^{(i)}$, becomes zero or sufficiently close to zero according to a pre-established tolerance criterion.

The general form of these equations for the i^{th} iteration is:

$$\underline{\underline{K}}^{(i-1)} \; \underline{\Delta X}^{(i)} = \underline{F}_o^{(i)} \qquad (3.58a)$$

$$\underline{X}^{(i)} = \underline{X}^{(i-1)} + \underline{\Delta X}^{(i)} \qquad (3.58b)$$

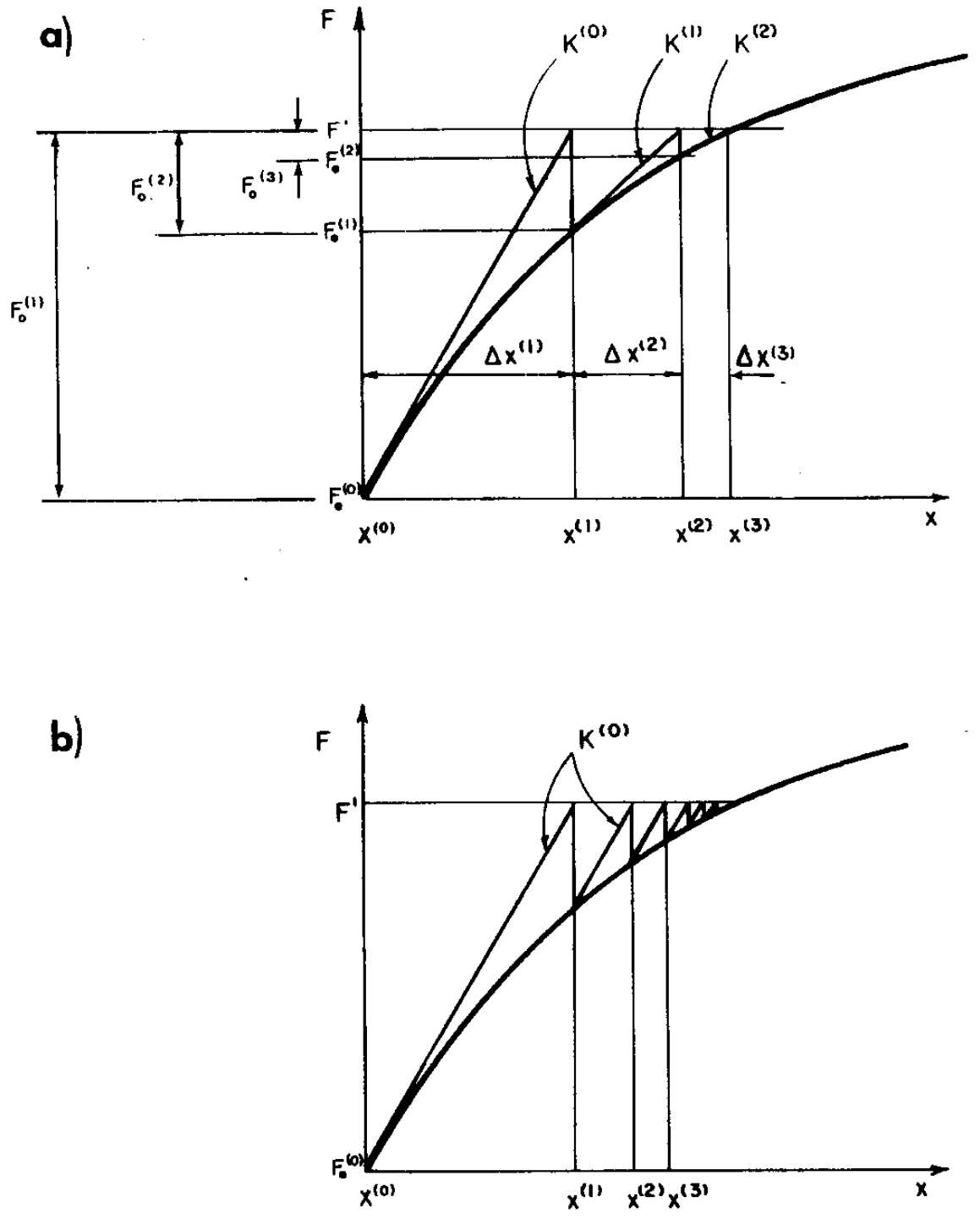


Figure 3.5 - Iterative Methods for 1-D Problem: a) Tangent or Newton Method; b) Modified Newton Method.

Using $\underline{X}^{(i)}$ the matrix $\underline{\underline{K}}^{(i)}$ is evaluated and,

$$\underline{F}_o^{(i)} = \underline{F}' - \underline{F}_e^{(i-1)} \tag{3.58c}$$

$$\underline{F}_e^{(i-1)} = \underline{\underline{K}}^{(i-1)}\underline{X}^{(i-1)} = \underline{\underline{K}}^{(i-1)}\underline{X}^{(i-2)} + \underline{\underline{K}}^{(i-1)}\underline{\Delta X}^{(i-1)} \tag{3.58d}$$

If the tolerance criteria is not satisfied the procedure is repeated. This method is called the *tangent procedure* (Desai and Abel, 1972) and is similar to the Newton-Raphson method for solving non-linear equations. This Newton-type iteration very effectively solves problems where large nonlinearities occur within the individual time intervals (Bathe, 1979). It is usually convergent, but if the initial guess $\underline{X}^{(0)}$ is not close to the solution, the problem may diverge (Zienkiewicz, 1977).

A modified version of this method, that requires less calculations, uses the initial value of $\underline{\underline{K}}$ throughout all iterations (Desai and Abel, 1972). This procedure usually requires a few more total iterations, but on the other hand, only one decomposition per time step is performed. For this alternative the iteration index for matrix $\underline{\underline{K}}$ in Eq. 3.58a is always (0). A graphical representation of this method for the case of one dimensional case is presented in Fig. 3.5b. The equations used in this method to solve the problem of Eq. 3.56 are:

$$\underline{F}_o^{(1)} = \underline{F} - \underline{\underline{K}}^{(0)}\,\underline{X}^{(0)} = \underline{F} - \underline{F}_e^{(0)} \tag{3.59}$$

and for the i^{th} iteration

$$\underline{\underline{K}}^{(0)} \underline{\Delta X}^{(i)} = \underline{F}_o^{(i)} \tag{3.59a}$$

$$\underline{X}^{(i)} = \underline{X}^{(i-1)} + \underline{\Delta X}^{(i)} \tag{3.59b}$$

$$\underline{F}_o^{(i)} = \underline{F} - \underline{F}_e^{(i-1)} \tag{3.59c}$$

$$\underline{F}_e^{(i-1)} = \underline{\underline{K}}^{(i-1)} \underline{X}^{(i-1)} = \underline{\underline{K}}^{(i-1)} \underline{X}^{(i-2)} + \underline{\underline{K}}^{(i-1)} \underline{\Delta X}^{(i-1)} \tag{3.59d}$$

Note that in Eq. 3.59d the matrix $\underline{\underline{K}}$ inside the summation is updated. However no assembly or decomposition of this matrix is necessary to perform this step, as Eq. 3.59d is calculated at the element level.

The tolerance criterion used in SWIM to determine when to stop the iterating is based on the *Eucledian norm* of the vectors involved. For some vector y this norm is:

$$||\underline{Y}|| = \left(\sum_{i=1}^{n} |y_i|^2 \right)^{1/2} \tag{3.60}$$

The calculation is said to converge at i^{th} iteration, and the computation stopped for that time step, if the following relationship is satisfied:

$$\frac{||\underline{\Delta X}^{(i)}||}{\max_t ||\underline{X}_t||} \leq \text{TOL} \tag{3.61}$$

where:

$\max_t ||\underline{X}_t||$ is the maximum Euclidean norm of the piezometric heads calculated during the solution, and

TOL is the relative tolerance used to measure equilibrium convergence.

It is also common to have in this type of numerical model a limit in the maximum number of iterations. In SWIM the default values for TOL and the maximum number of iterations are, respectively, 0.001 and 15.

The modified Newton solution procedure for non-linear problems has been adopted for the solution of the implicit time integration scheme of Eq. 3.53

$$(\frac{\underline{\underline{C}}^*}{\Delta t} + \underline{\underline{K}}_{t+\Delta t})\ \underline{X}_{t+\Delta t} = \underline{F}_{t+\Delta t} + \frac{\underline{\underline{C}}}{\Delta t}\ \underline{X}_t \qquad (3.53)$$

The matrix $\underline{\underline{C}}$ is assumed constant in time, while the matrix $\underline{\underline{K}}_t$ is updated every n time steps, $n \geq 1$. $\underline{\underline{K}}_t$ is decomposed into *linear* and *non-linear* parts:

$$\underline{\underline{K}}_t = \underline{\underline{KL}} + \underline{\underline{KNL}}_t \qquad (3.62)$$

where:

$\underline{\underline{KL}}$ is the linear part of the $\underline{\underline{K}}$ matrix, that belonging to a confined aquifer without an interface; it is only calculated once.

$\underline{\underline{KNL}}_t$ is the non-linear part of the $\underline{\underline{K}}$ matrix evaluated at time t, the only part that requires updating.

Table 3.1 outlines in detail the solution procedure for SWIM. At the

beginning of each time step a new approximation of the piezometric head $\underline{X}^{(0)}$ and the interface position are calculated using the results of the previous time step--a kind of prediction step. These initial estimates ($\underline{X}^{(0)}$) are refined by iteration until the solution converges. If the solution diverges, program execution ceases. The initial conditions for a transient problem must usually be consistent: that is, the initial states of the freshwater piezometric head and the interface (or saltwater piezometric head) must be compatible with each other and the initial boundary conditions, or the solution may diverge.

The system of simultaneous algebraic equations (e.g., Fig. 3.59a, or steps A2, B3 or B4d of Table 3.1) is solved by Gauss elimination using $\underline{\underline{L}}\ \underline{\underline{D}}\ \underline{\underline{L}}^{T}$ factorization, followed by reduction and back substitution of the flow vector each time $\underline{X}$ or $\underline{\Delta X}$ is calculated. The computer storage requirements are reduced due to the fact that only the elements below the matrix skylines are stored. Bathe and Wilson (pp. 249-258, 1976) present the details of the equation solver used. For larger systems the solution procedure may require an out-of-core solution, which is incorporated in SWIM.

Table 3.1

Step By Step Solution Procedure of SWIM

INITIAL CALCULATIONS

1 - Form linear conductivity matrix $\underline{\underline{KL}}$ and storage matrix $\underline{\underline{C}}$

2 - Calculate *effective linear conductivity matrix* $\underline{\underline{K}}^*$

$$\underline{\underline{K}}^* = \underline{\underline{KL}} + \frac{1}{\Delta t}\,\underline{\underline{C}}^*$$

3 - In a linear problem, triangularize $\underline{\underline{K}}^*$.

FOR EACH TIME STEP

A - Linear problem

1 - Compute the effective flow vector:

$$\underline{F}^*_{t+\Delta t} = \underline{F}_{t+\Delta t} + \frac{1}{\Delta t}\,\underline{\underline{C}}\,\underline{X}_t$$

2 - Solve for nodal points heads at time t + Δt:

$$\underline{\underline{K}}^*\,\underline{X}_{t+\Delta t} = \underline{F}^*_{t+\Delta t}$$

B - Non-linear problem

1 - For the first time step or if a new conductivity matrix $\underline{\underline{K}}$ is to be reformed, proceed as follows; otherwise go to Part 2 below. Calculate a new non linear part of $\underline{\underline{K}}$, $\underline{\underline{KNL}}$, using initial conditions for the first time step, or otherwise, using the solution from the previous time step. Obtain a new total effective conductivity matrix $\underline{\underline{K}}^*_t$:

Table 3.1 (Continued)

$$\underline{\underline{K}}^{*}_{t} = \underline{\underline{KNL}}_{t} + \underline{\underline{K}}^{*}$$

factor $\underline{\underline{K}}^{*}_{t}$

$$\underline{\underline{K}}^{*}_{t} = \underline{\underline{L}}\ \underline{\underline{D}}\ \underline{\underline{L}}^{T}$$

2 - Compute the effective flow vector:

$$\underline{F}^{*}_{t+\Delta t} = \underline{F}_{t+\Delta t} - \underline{\underline{K}}_{t}\ \underline{X}_{t}$$

where $\underline{\underline{K}}_{t} = \underline{\underline{KL}} + \underline{\underline{KNL}}_{t}$ (Eq. 3.62)

3 - Solve for nodal piezometric head increment $\underline{\Delta X}^{(0)}$ using latest $\underline{\underline{D}}$, $\underline{\underline{L}}$ factors. This is a predictor for $\underline{X}^{(0)}_{t+\Delta t} = \underline{X}_{t} + \underline{\Delta X}^{(0)}$ that is always calculated regardless of the number of iterations allowed.

$$\underline{\underline{L}}\ \underline{\underline{D}}\ \underline{\underline{L}}^{T}\ \underline{\Delta X}^{(0)} = \underline{F}^{*}_{t+\Delta t}$$

4 - If equilibrium solution sought, iterate for flow equilibrium and check for convergence. Initialize the iteration counter i = 0. Then:

(a) i = i + 1

(b) Calculate $(i-1)^{st}$ approximation to nodal point heads:

$$\underline{X}^{(i-1)}_{t+\Delta t} = \underline{X}_{t} + \underline{\Delta X}^{(i-1)}$$

$$\left(\frac{d\underline{X}}{dt}\right)^{(i-1)}_{t+\Delta t} = \frac{\underline{X}^{(i-1)}_{t+\Delta t} - \underline{X}_{t}}{\Delta t} = \frac{\underline{\Delta X}^{(i-1)}}{\Delta t}$$

Table 3.1 (Continued)

(c) Calculate the i^{th} out-of-balance flow:

$$\underline{F}_o^{(i)} = \underline{F}_{t+\Delta t} - \underline{F}_e^{(i-1)} - \underline{\underline{C}} \left.\frac{d\underline{X}}{dt}\right|_{t+\Delta t}^{(i-1)}$$

$$\underline{F}_e^{(i-1)} = \underline{\underline{K}}_{t+\Delta t}^{(i-1)} \underline{X}_{t+\Delta t}^{(i-1)} \qquad i > 1$$

(d) Solve for the i^{th} piezometric head increments:

$$\underline{\underline{L}}\ \underline{\underline{D}}\ \underline{\underline{L}}^T\ \underline{\Delta X}^{(i)} = \underline{F}_o^{(i)}$$

(e) Calculate new heads:

$$\underline{X}_{t+\Delta t}^{(i)} = \underline{X}_{t+\Delta t}^{(i-1)} + \underline{\Delta X}^{(i)}$$

(f) Iteration converges if

$$\frac{||\underline{\Delta X}^{(i)}||}{\max_t ||\underline{X}_t||} < TOL$$

- If solution converges: $\underline{X}_{t+\Delta t} = \underline{X}_{t+\Delta t}^{(i)}$ and go to part C
- If solution does not converge and if less than maximum number of iterations allowed: go to (a)
- Otherwise restart solution using a new conductivity matrix reformation interval and/or a smaller time step.

Table 3.1 (Continued)

C - Calculate Nodal Point Solution

(a) Heads: $\underline{X}_{t+\Delta t} = \begin{bmatrix} \phi_1^f \\ \phi_1^s \\ \vdots \\ \phi_n^f \\ \phi_n^s \end{bmatrix}$

(b) Interface depth:

$$\zeta_{t+\Delta t} = \frac{1}{\Delta\gamma} \cdot \begin{bmatrix} -\gamma^f & \gamma^s \end{bmatrix} \cdot \begin{bmatrix} (\underline{\phi}^f_{t+\Delta t})^T \\ (\underline{\phi}^s_{t+\Delta t})^T \end{bmatrix}$$

Chapter 4

TOE TRACKING ALGORITHM

4.1 Introduction

The previous chapters discuss the importance of tracking the toe in the seawater intrusion problem. The numerical aspects of this feature are presented and described in this chapter.

When thinking of seawater intrusion in aquifers, the term "toe" usually represents the intersection of the interface with the lower boundary, the bottom of the aquifer. However, there are situations as illustrated in Figure 4.1, where the interface also intersects the upper boundary. This occurs, for example, in a confined aquifer when the top confining layer is semi-pervious and the piezometric heads in both aquifer and confining layer are such that all the freshwater flows upwards through the aquitard and into the shallow aquifer. The Delta aquifer of Egypt offers and example (Wilson et al., 1979; Hashish et al., 1979). The resulting "upper toe" is treated by SWIM in the same way as the "lower toe". In this chapter when the term "toe" is mentioned it implies the lower toe; otherwise, the qualifier upper or lower, is used.

In the following sections different methodologies to track the toe are presented, with the emphasis on the algorithm employed by SWIM. This algorithm employs Gaussian quadrature points, used for the numerical integration over the elements, to determine the position of both the upper and lower toes. To evaluate the accuracy of this

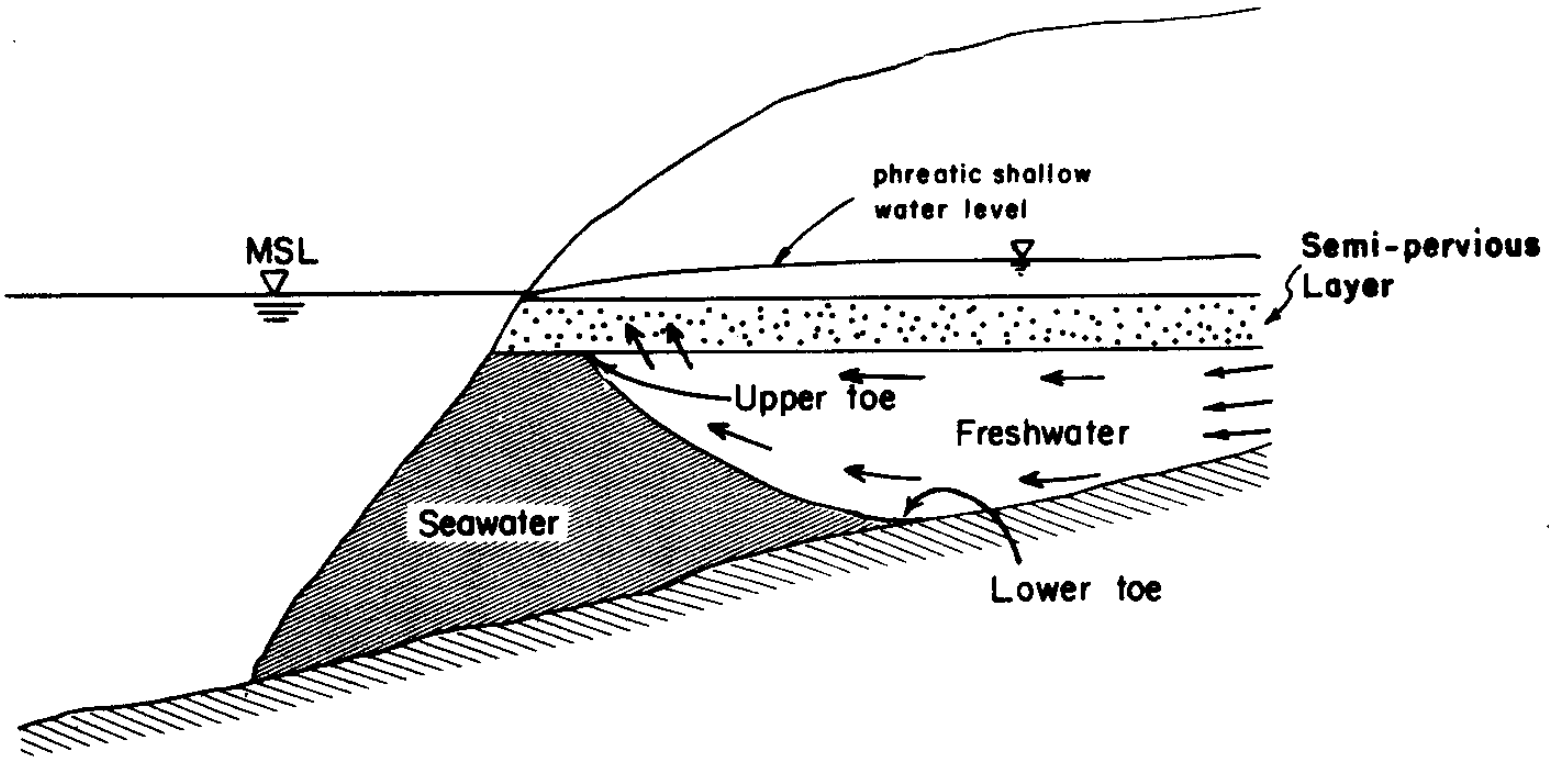


Figure 4.1 - Aquifer with an Upper and a Lower Toe

method, SWIM solutions are compared with the analytical solution of the gravitational segregation problem in porous media. For this situation the results of a sensitivity analysis are shown, to illustrate the dependence of this technique to various parameters. The chapter ends with the outline of some general guidelines for the use of the Gauss points to track the toe.

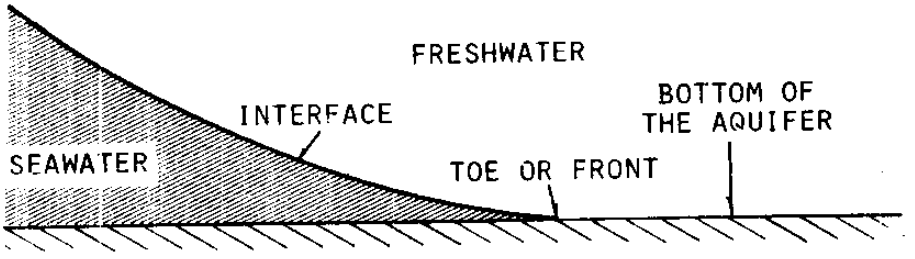
4.2 Different Methods to Track the Toe Movement

Consider the following techniques for tracking the movement of the toe, which is a "moving front" or a "moving boundary condition", for a finite element model:

1) ignore the position of the front inside individual elements,
2) mesh regeneration,
3) mesh displacement,
4) use of Gauss quadrature points.

To simplify the description in this section the locus of points defining the seawater wedge toe is called the *front*.

Consider the typical cross section of the interface near the toe shown in Figure 4.2, and the plan view of the toe with an hypothetical regular finite element grid represented in the same figure. *Ignoring the position of the front inside an element* means that the front can only be defined by the sides of the elements, as shown in Figure 4.3. Thus it is assumed that the front is never located in an intermediate position within the element. *Mesh regeneration* represents the front by regenerating a new finite element mesh at each time step, in such a way



SECTION A-A'

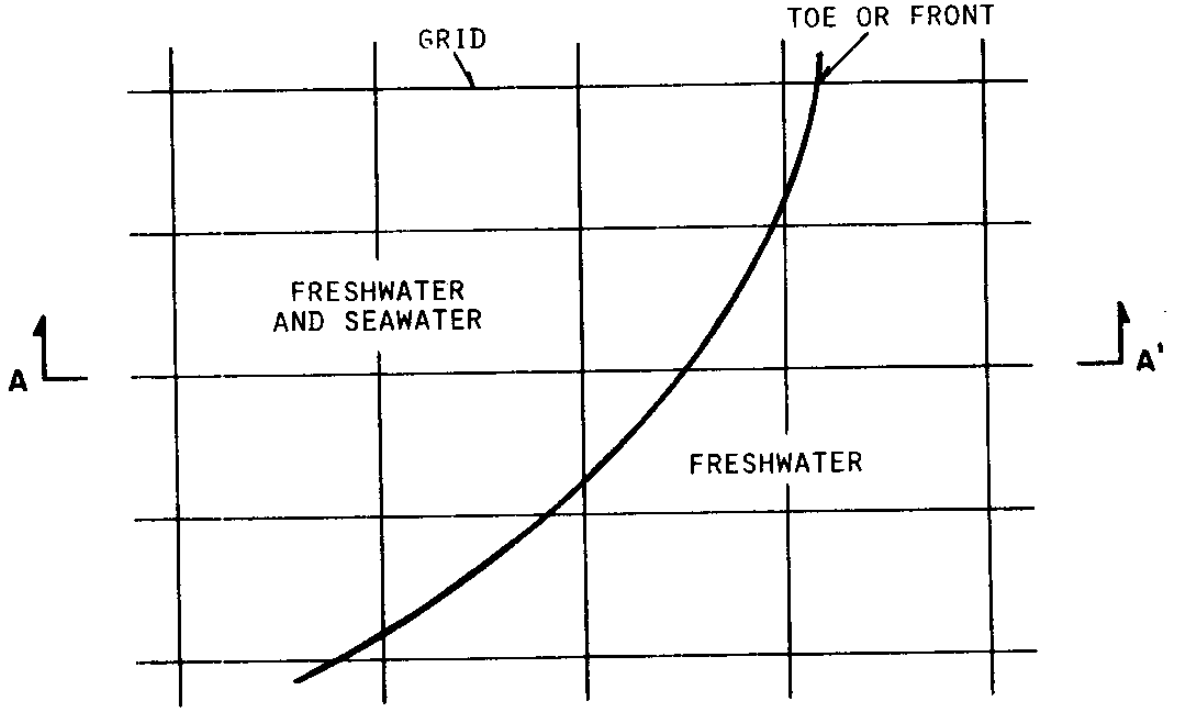


Figure 4.2 - Typical Plan View and Cross Section of the Interface Near the Bottom of the Aquifer

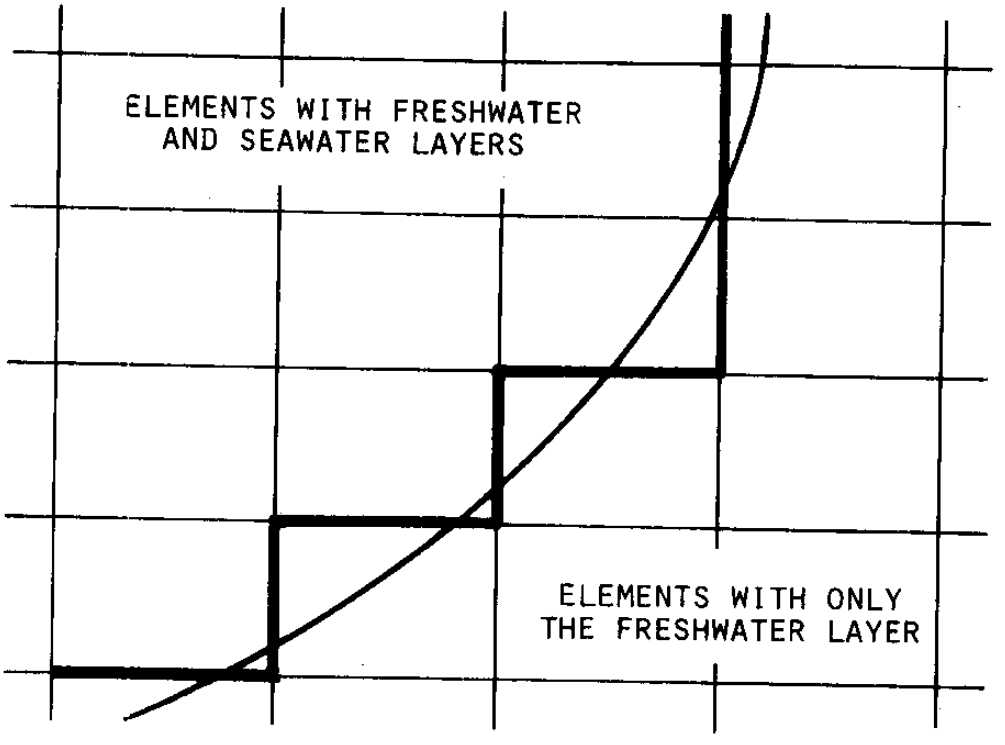


Figure 4.3 - Plan View of the Toe Location by a Model Ignoring the Toe Inside the Elements

that the front is always defined by nodes. This technique is relatively easy to implement in 1-D problems, as demonstrated by Shamir and Dagan (1971). But it becomes rather difficult to implement for the general 2-D problem with an irregular frontal shape, due to the logic needed to properly set up a new mesh and the computational burden of reforming all matrices. Figure 4.4 illustrates one simple type of mesh regeneration, in which the nodes near the front are displaced in only one direction, to match the front. It has been used, for example, by Lefebvre et al. (1974) in a petroleum reservoir model. The *mesh displacement method* is actually a type of mesh regeneration. It uses a series of fixed node points at the front. At each time step these nodes and the entire mesh are displaced with the front. This method becomes very complicated and costly to use in the general 2-D field problem, where the front does not move in a regular fashion, and when interest is focussed on the entire domain, rather than the zone moving with the front. It is often used for frontal displacements in Stefan type problems (see Fisher and Medland, 1974; and Crank, 1974).

The fourth method uses *Gauss quadrature points* to indirectly track the front position. In SWIM, as in most finite element models, the spatial integrals of the discretized equations are evaluated numerically using Gauss quadrature. Each integral is approximated by a properly weighted sum of values of the integrand taken at suitable locations distributed within each element. A Gauss quadrature with n points gives an exact representation of a polynomial of order (2n-1).

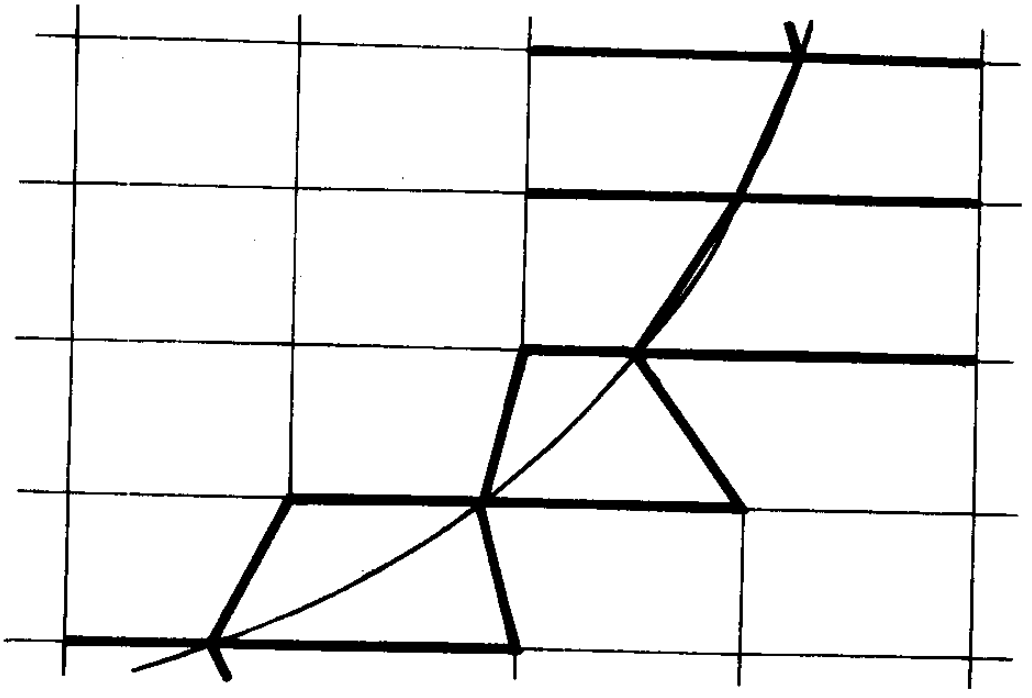


Figure 4.4 - Plan View of the Toe Location by a Mesh Regeneration Model

It is usually suggested that for the evaluation of the conductivity matrix, in 2-D elements, the optimum order of integration is 2 × 2, or 3 × 3 if the element is highly distorted or if saturated thickness varies over quadratic elements; for storage and third type boundary matrices a 3 × 3 order is used (Bathe, 1976, 1979). The term m × m for 2-D elements refers to the number m of Gauss points oriented in each of two mutually orthogonal directions of the local coordinate system r, s.

When using the Gauss points to track the front, the interface position is determined at each Gauss point and compared with the elevation of the bottom of the aquifer at the same location, in order to determine the presence of a seawater layer. In other words two sets of Gauss points are implicitly defined for the conductivity matrix: one inland of the front where there is no seawater; the other seaward of the front where there are two layers, one of freshwater the other of seawater (see Figure 4.5). For computational purposes the model recognizes that the front lies between these two sets of points, although the actual front position is never directly calculated and used in the computation. The front is directly located only for output purposes, for which various post-processing techniques are discussed later. When the front is inside an element the quadrature order for the conductivity matrix is usually increased to 4 × 4, as shown in Figure 4.5, to better define the front, especially for large and/or distorted elements. This technique is clearly more accurate than ignoring the front inside the element. It is also less expensive and

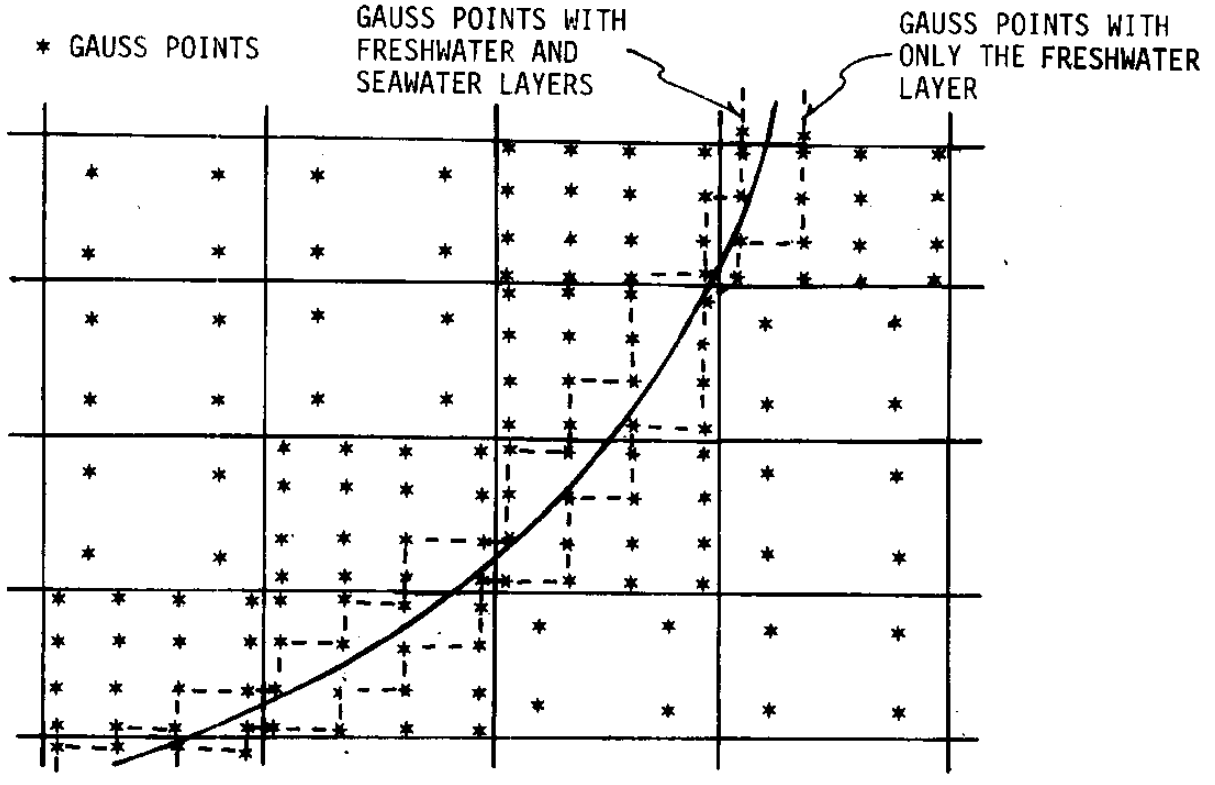


Figure 4.5 - Plan View of the Toe Location by a Model Using the Gauss Quadrature Points and Linear Elements (SWIM)

as accurate as mesh regeneration or displacement. In the following sections the details of this technique are discussed. A similar approach was used by Bathe and Khoshgoftaar (1977) to determine the free surface position in a vertical 2-D model for seepage problems.

4.3 Use of Gauss Points to Track the Toe

SWIM is a 2-D model in the horizontal plane. However, for the sake of simplicity of the figures in this section, 1-D examples are used without loss of generality. To judge the merits of the technique proposed, SWIM was applied to a simple problem with an analytical solution: one dimensional gravitational segregation in a porous medium.

4.3.1 Gravitational Segregation Problem

Suppose that two fluids of different specific weight, γ^f and γ^s, occupy the pore space of an horizontal confined aquifer of height b as shown in Figure 4.6a. The fluids are separated by an immiscible interface, that is initially oriented in the vertical direction. As time elapses the heavier fluid γ^s displaces the lighter fluid γ^f (see Figure 4.6b). The interface is described by the simple expression (Gelhar et al., 1972)

$$b^s = \frac{1}{2}\left(b + \frac{x}{\sqrt{\tau}}\right) \qquad (4.1)$$

in which:

b^s = the thickness of seawater below the interface

$\tau = \frac{K}{n}\frac{\Delta\gamma}{\gamma}\frac{t}{b}$ = dimensionless time

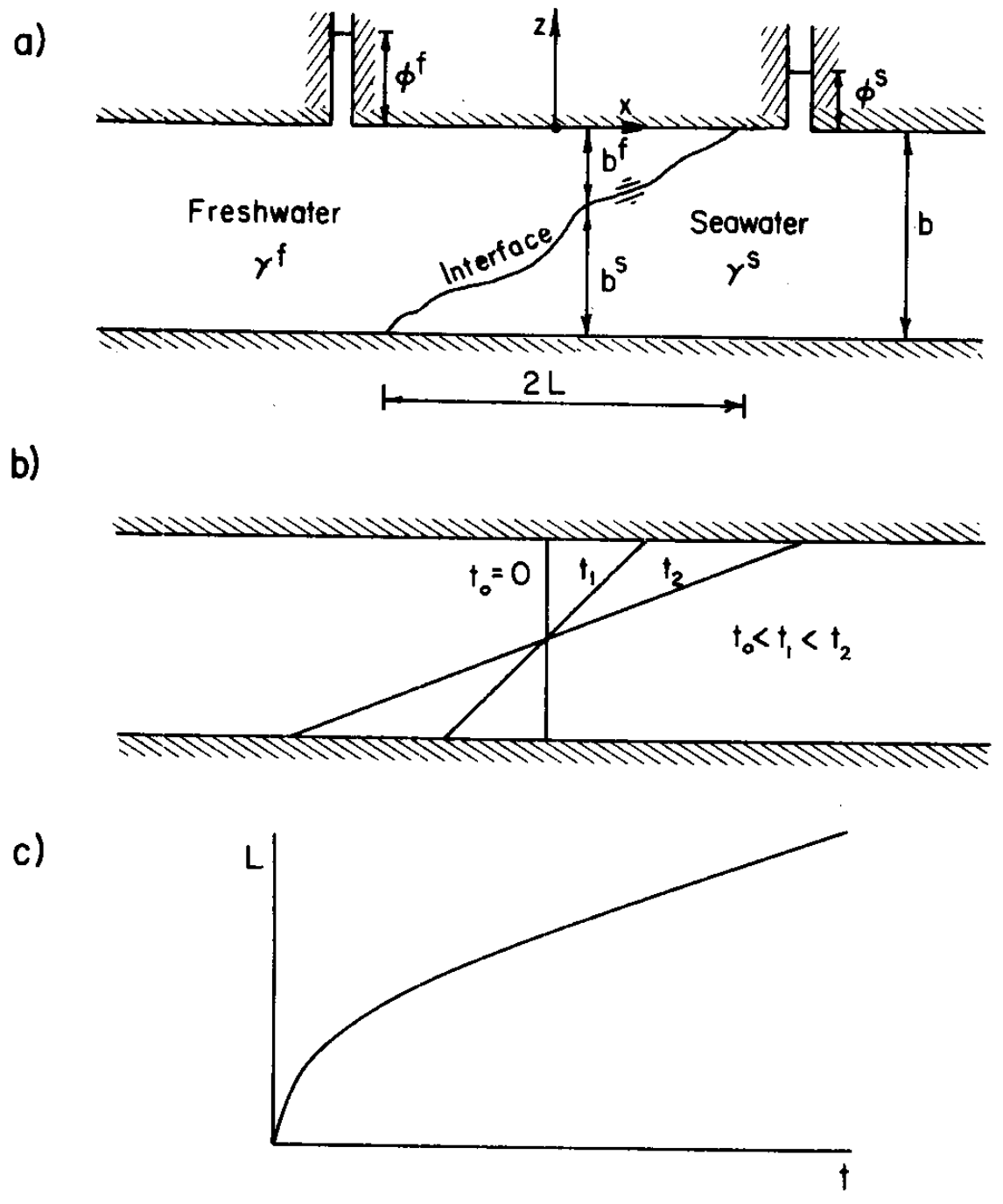


Figure 4.6 - The Gravitational Segregation Problem. a) Definition Sketch; b) Interface Movement; c) Toe Movement.

K = permeability

n = porosity

t = time

$\Delta\gamma = \gamma^s - \gamma^f$

$\gamma = \gamma^f$

Eq. 4.1 says that the interface shape is a straight line. The location of the interface toe L, either lower (-) or upper (+), is described by:

$$L = \pm b\sqrt{\tau} \tag{4.2}$$

which indicates that the rate of displacement decreases with time, as the density gradients are damped out, as also shown in Figure 4.6c.

For the examples of this chapter the following parameter values have been used for the analytical solution of the gravitational segregation problem:

K = 39.024 m/day

$\gamma^f = \gamma = 1.0$ g/cm^3

$\gamma^s = 1.025$ g/cm^3

$\Delta\gamma = 0.025$

$n = 0.3$

$b = 10.0$ m

These values are identical to those used by Shamir and Dagan (1971) for their finite difference numerical simulation of this problem using mesh regeneration. In the SWIM numerical simulation of this problem:

$K^f = K = 39.024$ m/day

$K^s = 40.0$ m/day

4.3.2 Description of the Technique

A vertical cross section of the lower part of the interface of the gravitational segregation problem is represented in Figure 4.7a, along with a one dimensional finite element grid. For the Gauss points seaward of the toe, to the right in Figure 4.7a, the transmissivity of the seawater is given by the product of the seawater permeability, K^s, and the seawater thickness, b^s, which is the distance between the interface and the bottom of the aquifer. For the Gauss points inland of the toe, to the left in Figure 4.7a, the seawater transmissivity is zero, because seawater does not exist there. However, in order to preserve the positive definitness of the conductivity matrix it is necessary to assign a small positive value for the seawater transmissivity at these points. This procedure is equivalent to assuming that inland of the toe there is a very thin seawater layer, underneath the freshwater, and that the permeability of this layer is very small compared with the usual K^s, say 10^5 times smaller. That is, the permeability follows a spatially non-linear rule, as shown in Figure 4.7b. The rule is time dependent, because the discontinuity point, at the front, changes position in time. In summary, this methodology assumes that there are two layers, one of freshwater and the other of seawater, over the entire aquifer domain, or at least over the area subject to seawater intrusion. Inland of the toe the transmissivity of the seawater layer is for practical purposes zero and does not affect the solution.

The finite element method is an integrated method and thus tends

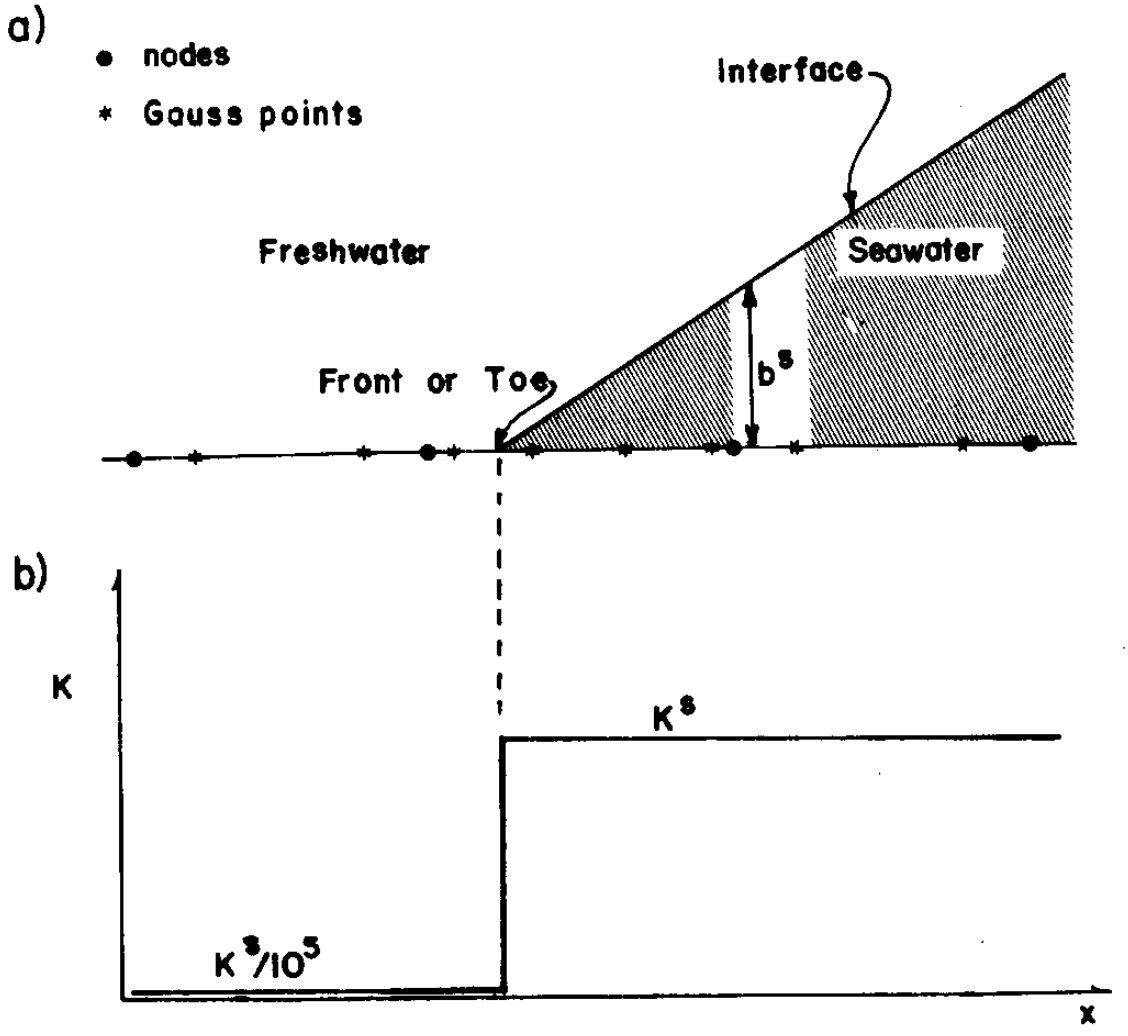


Figure 4.7 - Toe Representation Using Gauss Points:
a) Toe Location in a Finite Element Grid;
b) Permeability Variation at the Toe.

to smooth sharp transitions, such as the one present at the toe. If the very thin layer (with thickness b_e) of seawater inland of the toe is considered above the bottom of the aquifer, as shown in Figure 4.8a, the smoothing of the interface at the toe zone will result in an acceleration of the front. On the other hand if the same layer of seawater is now considered under the bottom of the aquifer, the same smoothing occurs but now the location of the front is determined by the intersection of the interface with the bottom (see Figure 4.8b) and the acceleration effect disappears. However, if the thickness of this layer of seawater below the bottom becomes too large, the role of the storage in this layer becomes important and a deceleration is experienced by the front, as illustrated in Figure 4.8c. The effect of using the seawater layer above or below the bottom of the aquifer does not affect the calculation of the position of the interface elsewhere; as a matter of fact for small values of the thickness of this layer the position of the interface at the second or third node seaward of the toe is practically unchanged. Figure 4.9 schematically represents the interface for the gravitational segregation problem as used in the SWIM code, as well as the rules of spatial variability of permeability and transmissivity for both freshwater and seawater. All simulations described in this report are based on an extra-thickness below the bottom of the aquifer, as shown in Figures 4.8b and 4.9.

Figure 4.10 presents a flow chart illustrating the use of this technique in SWIM. This flow chart does not pretend to be exhaustive in detail. Rather, it synthesizes what has been said about the technique and connects it to the description of SWIM given in Table 3.1.

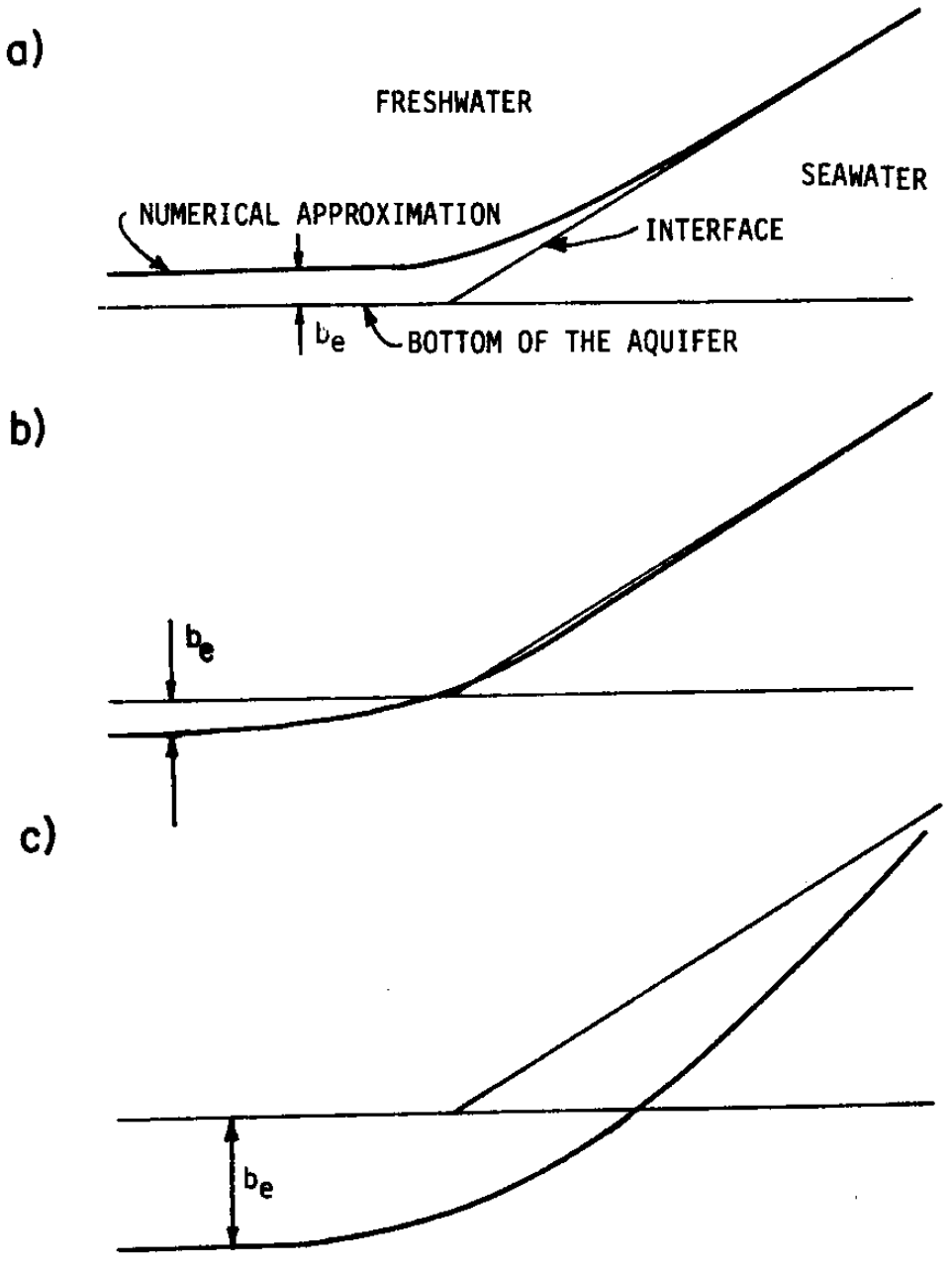


Figure 4.8 - Extra-Layer Options: a) Thin Layer Above the Aquifer's Bottom; b) Thin Layer Below the Aquifer's Bottom; c) Thick Layer Below the Aquifer's Bottom.

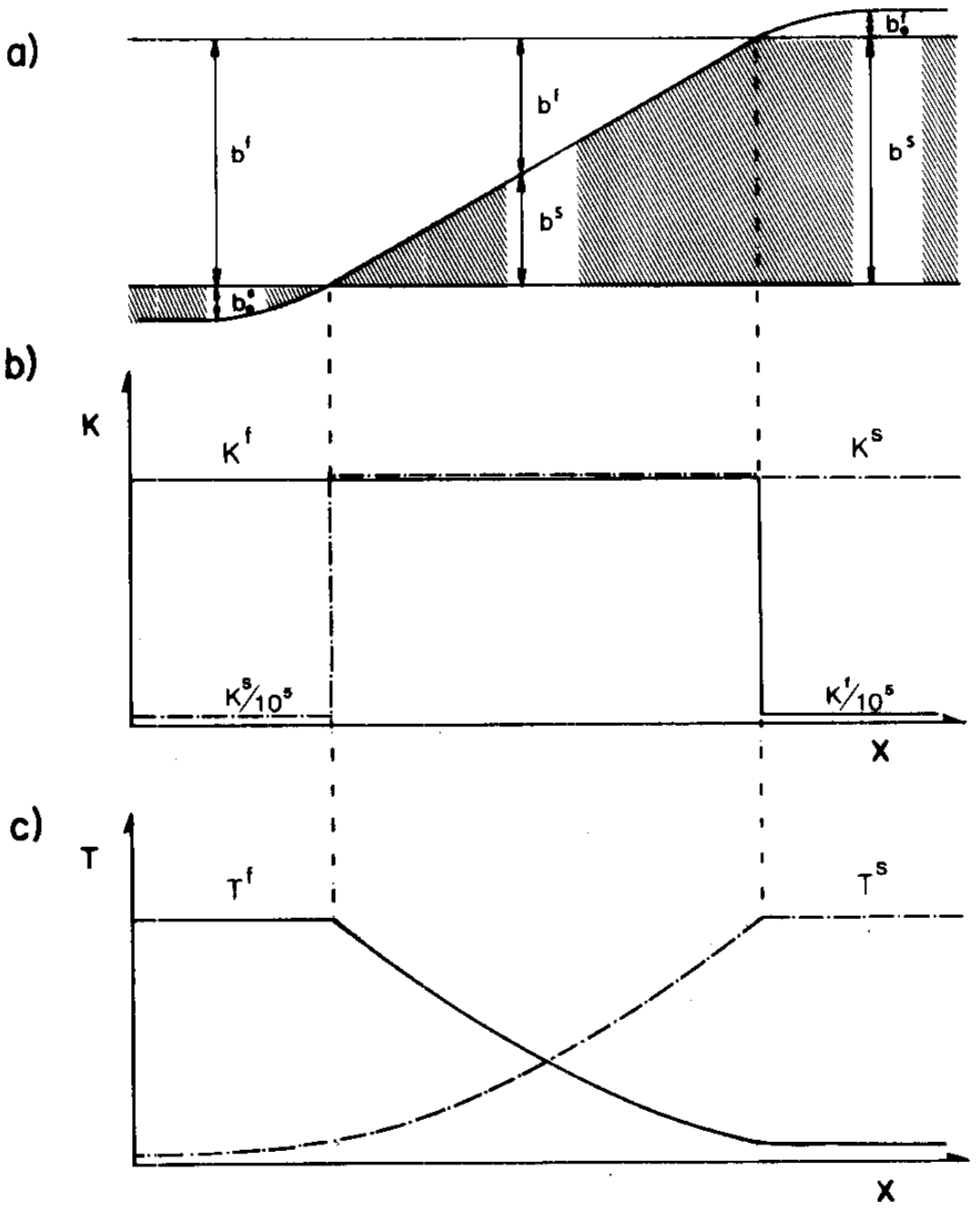


Figure 4.9 - SWIM Representation of the Gravitational Segregation Problem
a) Representation of the Interface and Toe;
b) Permeability Variation;
c) Transmissivity Variation ($c = a \times b$).

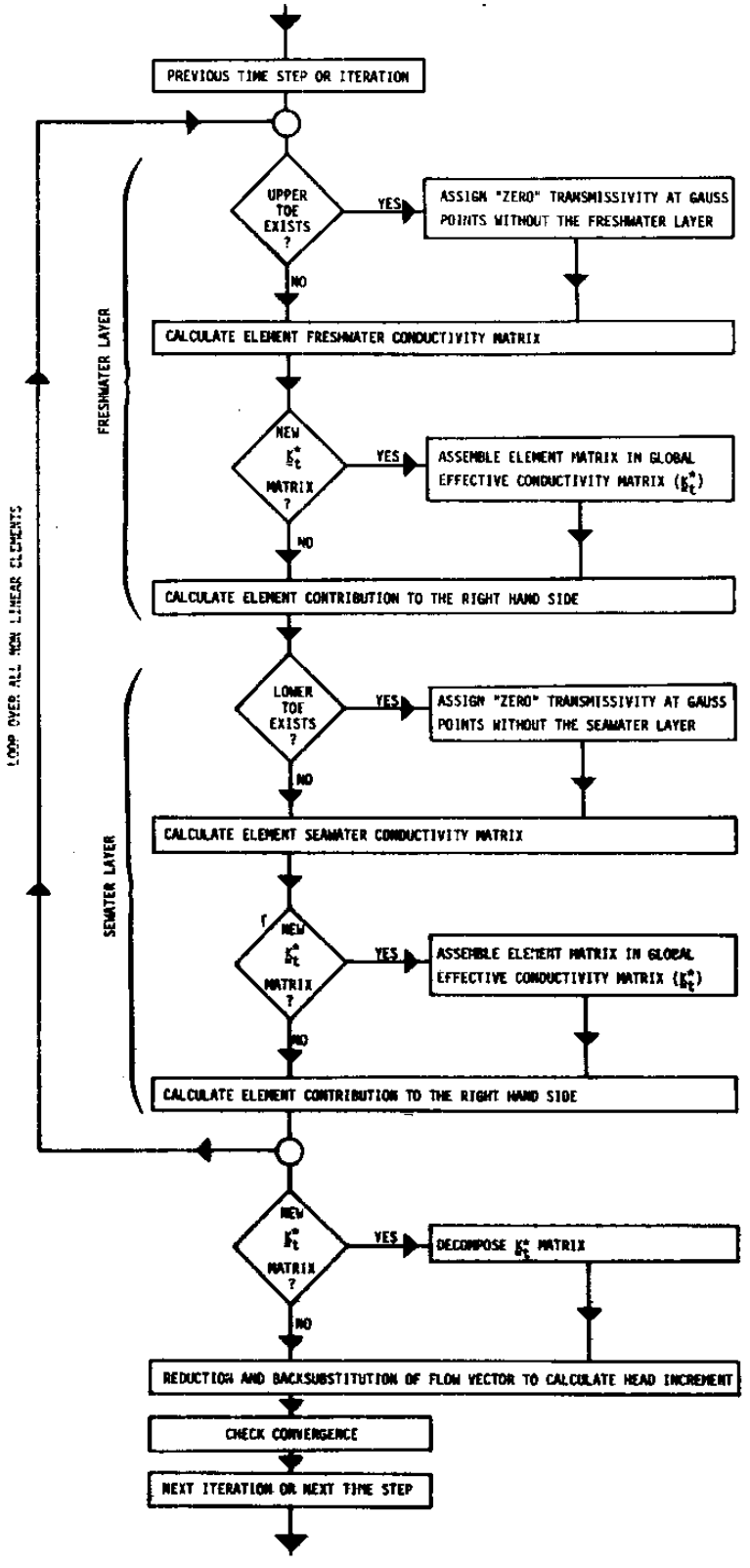


Figure 4.10 - Flow Chart of the Calculation of the Non-Linear Conductivity Matrix in SWIM

In the follwoing sections the results obtained for the numerical simulation of the gravitational segregation problem for different space and time discretizations, as well as for different values of the "extra-thickness" and its permeability, are presented. This presentation only concerns the immediate vicinity of the toe, because away from it the solution is not very sensitive to the various options considered.

4.3.3 Numerical Simulation

The analytical solution of the gravitational segregation problem is for an interface that is initially vertical. In a numerical model this initial condition requires a very detailed grid around the origin, and a very small time step in order to capture the early interface movement. Since this early movement is of no particular interest here, the initial condition (see Appendix C) used in the numerical simulation was taken as the position of the interface position after $t = t_o \simeq 7.87$ days ($\tau = \tau_o = 2.56$). At this time the positions of the upper and lower toe are, respectively, $L/b = +1.6$ and $L/b = -1.6$, or $L = \pm 16.0$ m. The simulation results obtained using a 136 m long regular grid, consisting of 4-node linear elements with a spacing of $\Delta x = 4.0$ m ($\Delta x/b = 0.4$) in the longitudinal direction (similar to, but longer than, grid 1 of Figure 4.18), are presented in Figures 4.11 and 4.12. The parameters used for this simulation were:

$n = 0.3$

$K^f = 39.024$ m/day

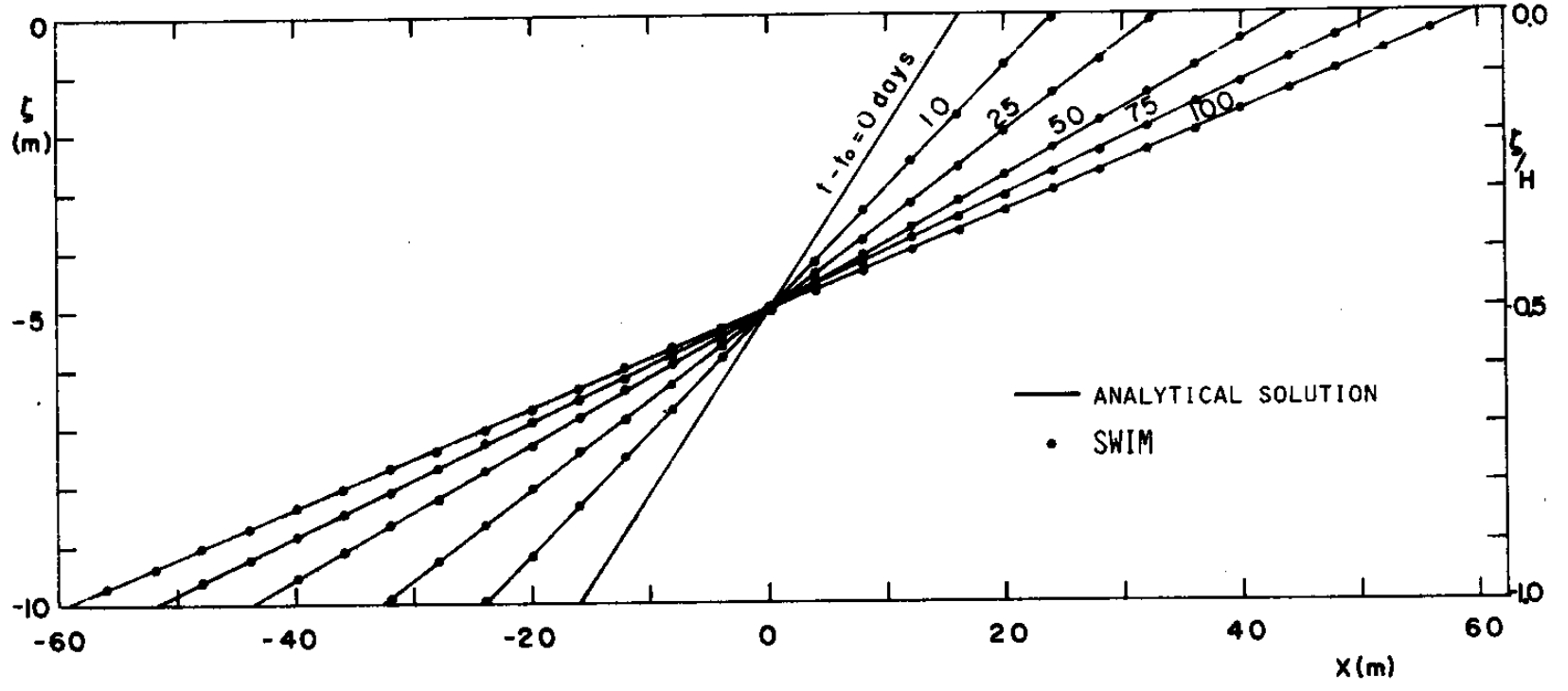


Figure 4.11 - Gravitational Segregation Problem, Comparison Between Analytical and SWIM Solutions for the Interface

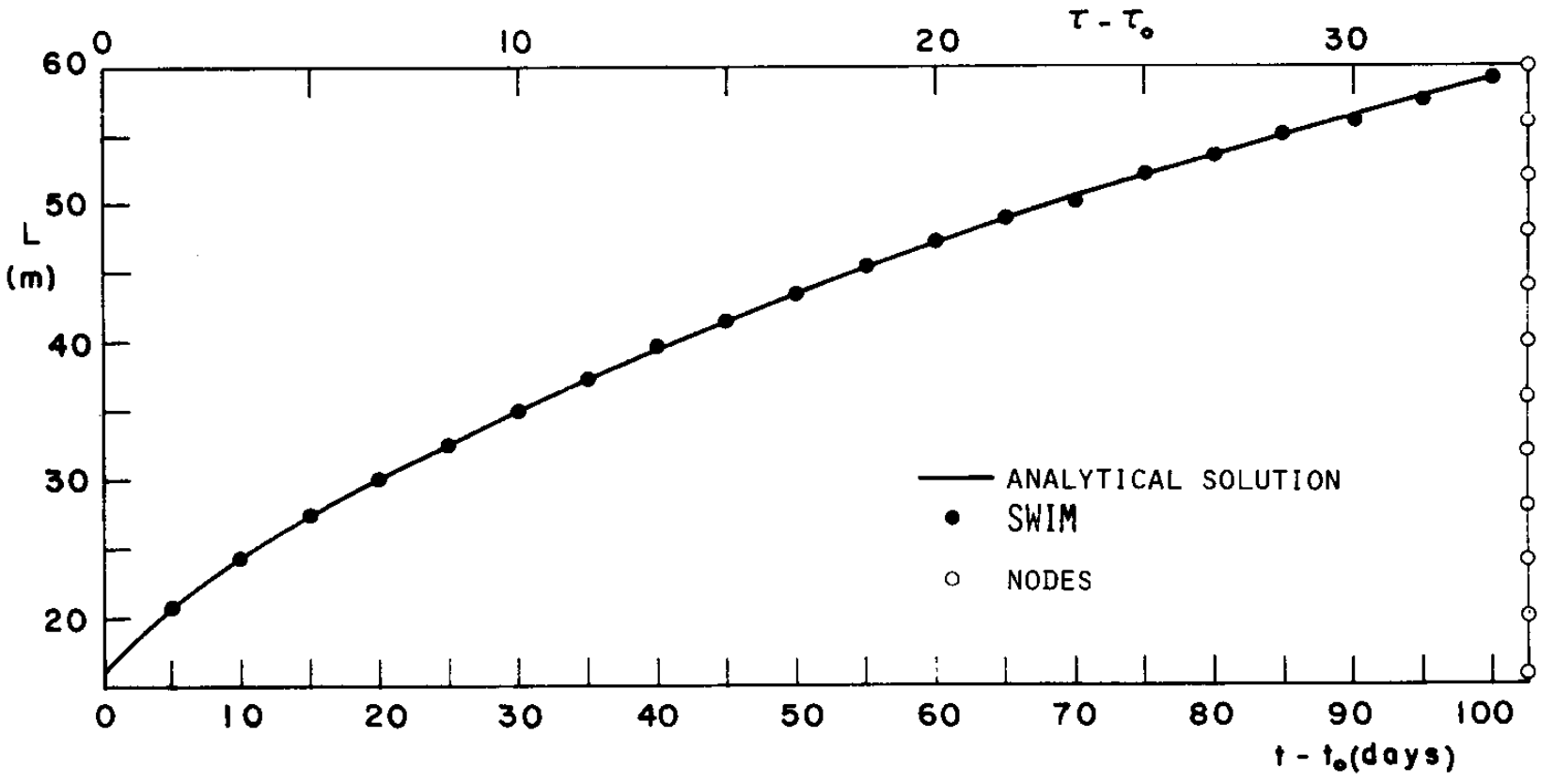


Figure 4.12 - Gravitational Segregation Problem, Comparison Between Analytical and SWIM Solutions for the Toe Movement (100 days)

K^s = 40.0 m/day

Δt = 1.0 day ($\Delta\tau$ = 1/3)

First type boundary conditions for both freshwater and seawater phases were specified at the two nodes at the far left of the grid (see Appendix C). A no flow condition was used elsewhere. In Figure 4.12 the toe location is shown only every fifth time step; the locations of node points are given along the right-hand side vertical axis.

To evaluate the toe position, as shown in Figure 4.12, SWIM results were post processed. Different alternatives were available for this purpose, among them the use of interpolation functions and extrapolation techniques. Since both the interface and interpolation functions were linear, a linear extrapolation using the interface elevation at the two nodes inland of the toe was used. This technique was also used when a quadratic grid was employed. Section 4.3.5 further discusses the subject of locating the toe for output purposes.

4.3.4 Sensitivity Analysis

The factors that can influence the solution for the interface, and particularly the "location" of the toe, can be divided in two groups: factors intrinsic to the toe "tracking" method such as the permeability and thickness of the extra-layer, the number of Gauss points, and the post-processing method of toe location; and factors extraneous to the method such as the space discretization, the time step, the number of time steps between updating the conductivity matrix, and the use of a consistent or lumped storage matrix.

Comparison of the solution sensitivity to these various factors is made relative to the "basic simulation", the results of which have already been shown in Figures 4.11 and 4.12. The specifications for this simulation, and all other sensitivity simulations unless otherwise noted, are:

-Grid	Grid 1 of Figure 4.18
-Gauss points in toe elements	4×4
-Initial conditions	$t_o = 7.87$ days, $\tau_o = 2.56$
-Time step	$\Delta t = 1$ day, $\Delta\tau = 1/3$
-Permeability of the extra-layer	$10^{-5}K$, where K is the corresponding aquifer permeability, K^f or K^s
-Thickness of extra-layer	$b_e = 10^{-3}b$, where b is the aquifer thickness
-Storage Matrix	Consistent (not tri-diagonal)
-Updating of Effective Conductivity Matrix	Every time step
-Method of location of the toe	Extrapolation

The sensitivity analysis examines the first 20 days ($\tau-\tau_o = 20/3$) after the initial conditions for the basis of comparison and, therefore, a slightly shorter grid (100 m) was used. Furthermore, since most of the simulations reproduced the interface shape of Figure 4.11 accurately, the comparison concentrates instead on the identification of the toe location from nodal values of the interface elevation, as in Figure 4.12. The extrapolation method of identification (see Section 4.3.5) is used unless otherwise noted. The first 20 days of Figure 4.12 are reproduced at a larger scale in Figure 4.13. Once again the node

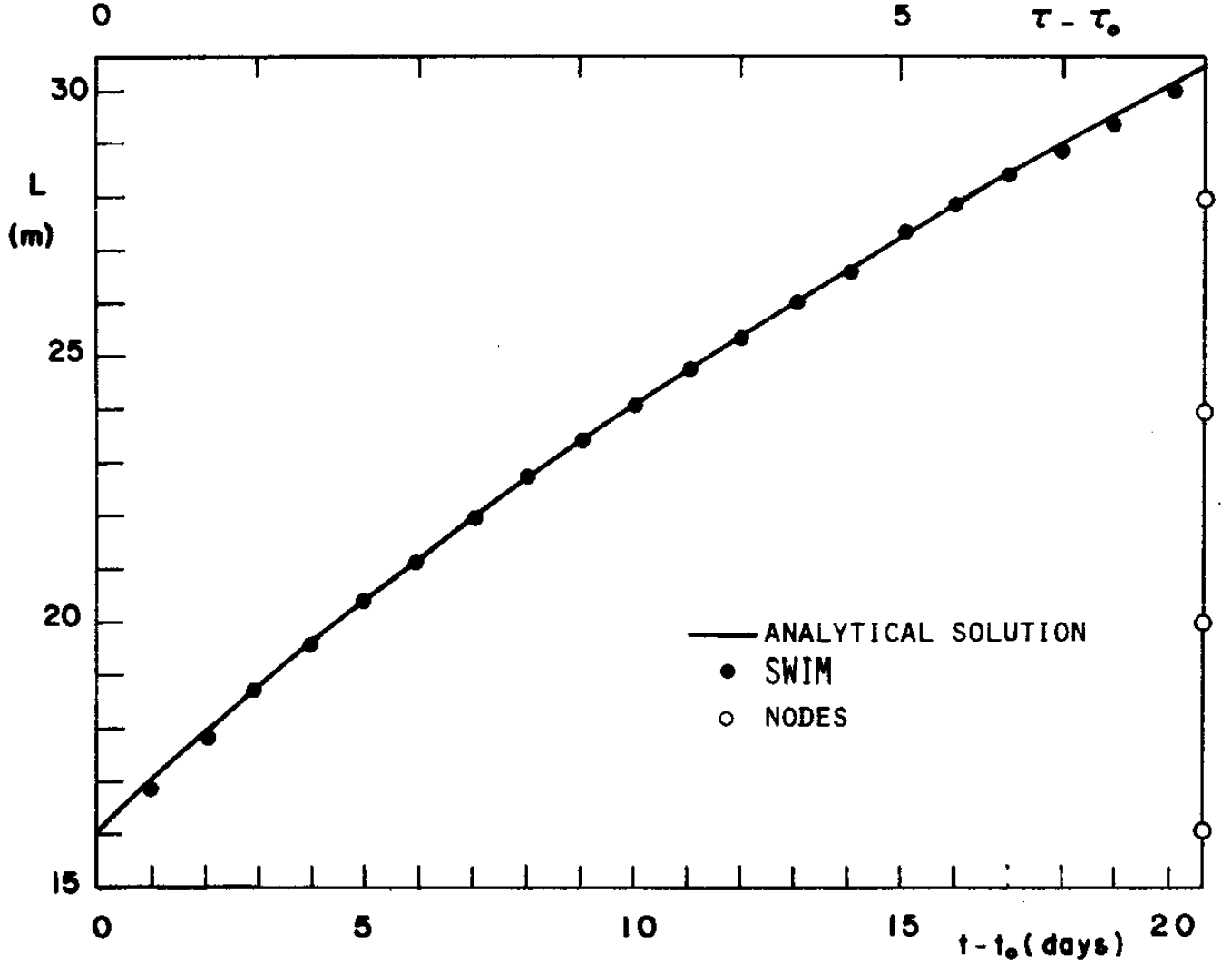


Figure 4.13 - Gravitational Segregation Problem, Comparison Between Analytical and Swim Solutions the Toe Movement (20 days)

points are located on the right-hand side vertical axis of the figure, as they are for most similar figures in this chapter.

4.3.4.1 Sensitivity to the "Extra-Layer" Algorithm

The extra-layer must be thick enough to absorb the smoothing typical of the finite elements near the toe. However, this thickness should not be so large as to lead to spurious solutions. The permeability of this extra-layer (K_e) should also be kept small compared to the aquifer permeability (K) to avoid significant flow through this layer. At first guess a reasonable order of magnitude for the ratio

$$\frac{b_e}{b} = \frac{\text{thickness of the extra-layer}}{\text{aquifer thickness}}$$

of 10^{-3} seems acceptable. Values ranging from 10^{-1} to 10^{-4} were tested. The permeability ratio (K_e/K) was varied from 10^{-3} to 10^{-8}.

No significant variation in the solution was found for the interface position, and consequently for the toe location, for different permeability ratios. For values smaller than 10^{-5} the results were identical up to the fifth significative digit. The value of 10^{-5} for the ratio of permeabilities (K_e/K) was adopted as a standard.

For the larger values of the extra-layer thickness ratio (b_e/b) the movement of the toe was significantly slowed, due to storage effects in the extra-layer and due to the smaller permeability that seems to decelerate the toe movement. The best results were found for the ratios of 10^{-3}. The results for the ratios of 10^{-2}, 10^{-3} (the basic simulation) and 10^{-4} are plotted in Figure 4.14.

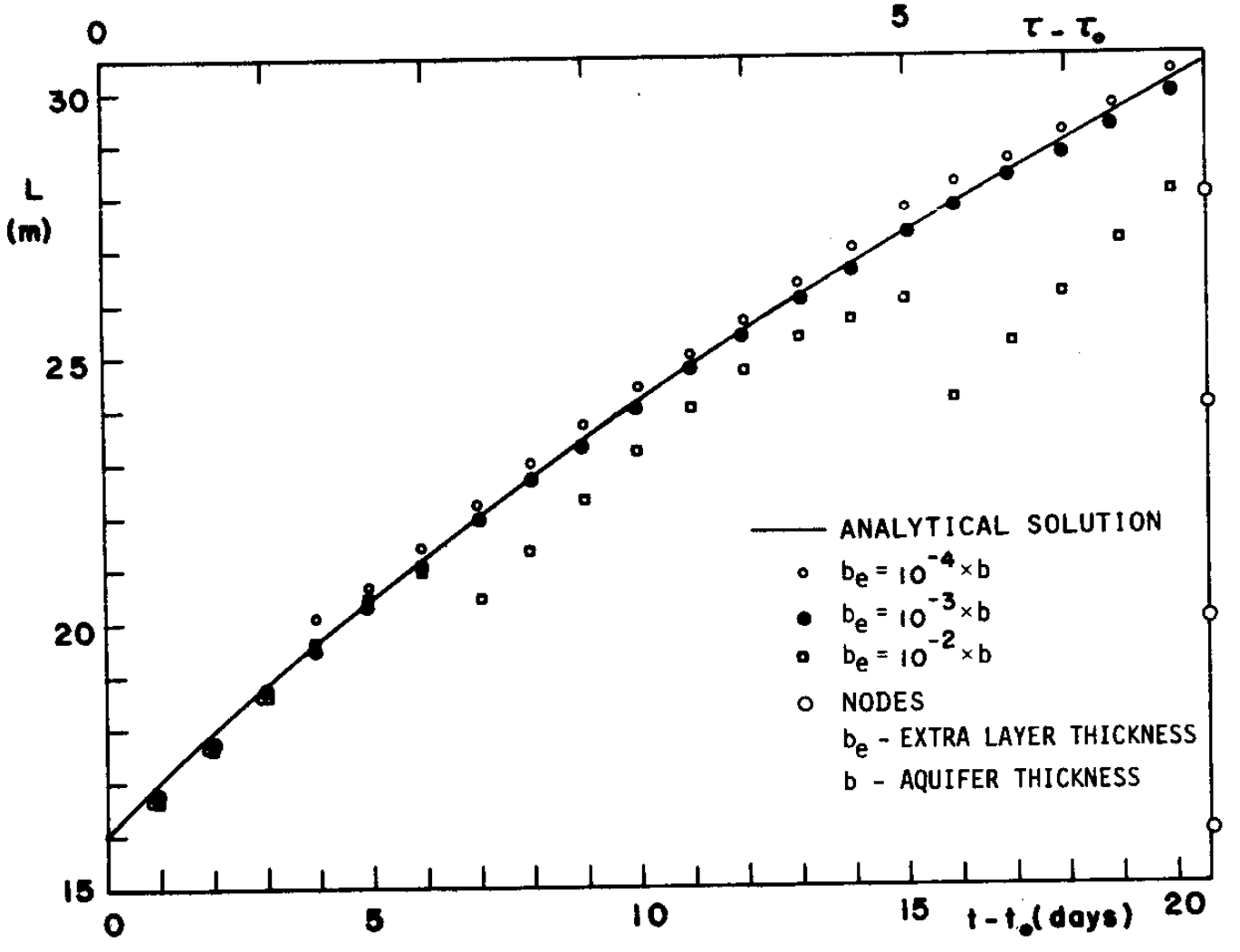


Figure 4.14 - Sensitivity of the Toe Movement to the Extra-Layer Thickness

For the ratio of 10^{-4} the effects of a slight acceleration appear, as one would expect if no extra-layer existed. However, this acceleration disappeared when a time step four times smaller was used (see Figure 4.15); the results of this simulation are practically equivalent to those of the basic run, indicating some dependence between the time step and the thickness of the extra-layer. Later, it is shown that the thickness of the extra-layer, the time step and the space discretization are related; however, no explicit expression for this relationship has been found.

Analyzing the toe movement in Figure 4.14 for the ratio $b_e/b = 10^{-2}$ reveals a backward jump between $t-t_o$ = 6 and 7 days and, again, between 15 and 16 days. These jumps are attributed to the large extra-layer thickness and to the method of calculation of the toe location. For $t-t_o$ = 5 or 6 days the linear extrapolation uses the nodal interface elevation at x = 12 and 16 m, because at x = 20 m the model has failed to recognize the presence of seawater yet. Nevertheless, the extrapolation projects the toe position to fall between x = 20 and 24 m. At $t-t_o$ = 7 days the interface at x = 20 m rises above the bottom of the aquifer, and the nodal points used to locate the toe shift to x = 16 and 20 m. However, due to the small difference of elevation between the bottom of the aquifer and the interface at x = 20 m, the extrapolation excessively underpredicts the toe location, originating the backward jump. Later, for $t-t_o$ > 7 days, the toe begins to recover until it reaches the next element boundary. Then the same thing happens again.

This behavior appears every time the toe crosses an element bound-

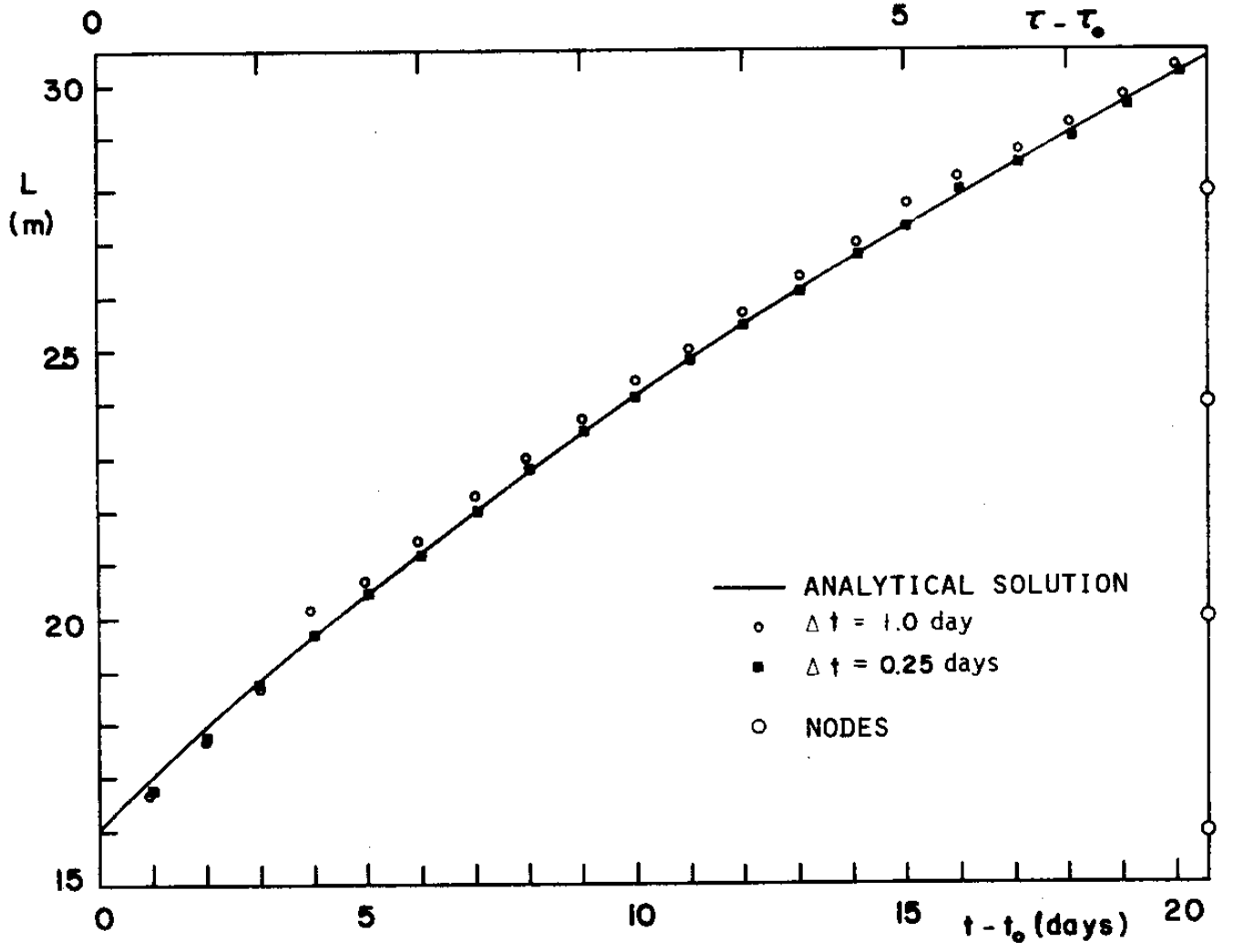


Figure 4.15 - Sensitivity of the Toe Movement to the Time Step Size for a Small Extra-Layer Thickness ($b_e = 10^{-4} b$)

ary and a "large" extra-layer thickness and/or large grids are used. For the basic simulation a similar phenomenon was observed when an element boundary was crossed, although it was of no significance. In that case the rate of toe movement was reduced slightly as the boundary was crossed, resulting in a minor underprediction of toe position for that time step. The prediction recovered in succeeding time steps. The situation can be observed in the last few time steps of Figure 4.13.

It must also be mentioned that the toe location error does not vary linearly with the extra-layer thickness. The solution for a thickness ratio $b_e/b = 5 \times 10^{-3}$ lies between the solutions for ratios of 10^{-2} and 10^{-3}, but much closer to the solution for 10^{-3} than to the one for 10^{-2}.

The interface position inland of the toe sometimes shows spatial oscillations due to the abrupt change of gradients, as illustrated in Figure 4.16. For example, an extra-layer thickness of $b_e = 10^{-3}b$ leads to oscillations above the bottom of the aquifer (see Figure 4.16b) for the first 72 days of simulation. But for $b_e = 5 \times 10^{-3}b$ the oscillations never appear above the bottom of the aquifer (see Figure 4.16c), indicating that the extra-layer has "absorbed" them. The oscillations are of minor relevance to the effectiveness of the algorithm and are typical of numerical methods. The extra-layer thickness for the basic run was selected to obtain the best results for interface and toe location, using the value of $b_e = 10^{-3}b$, even though this meant some minor oscillation of the interface inland of the toe. The amplitude of

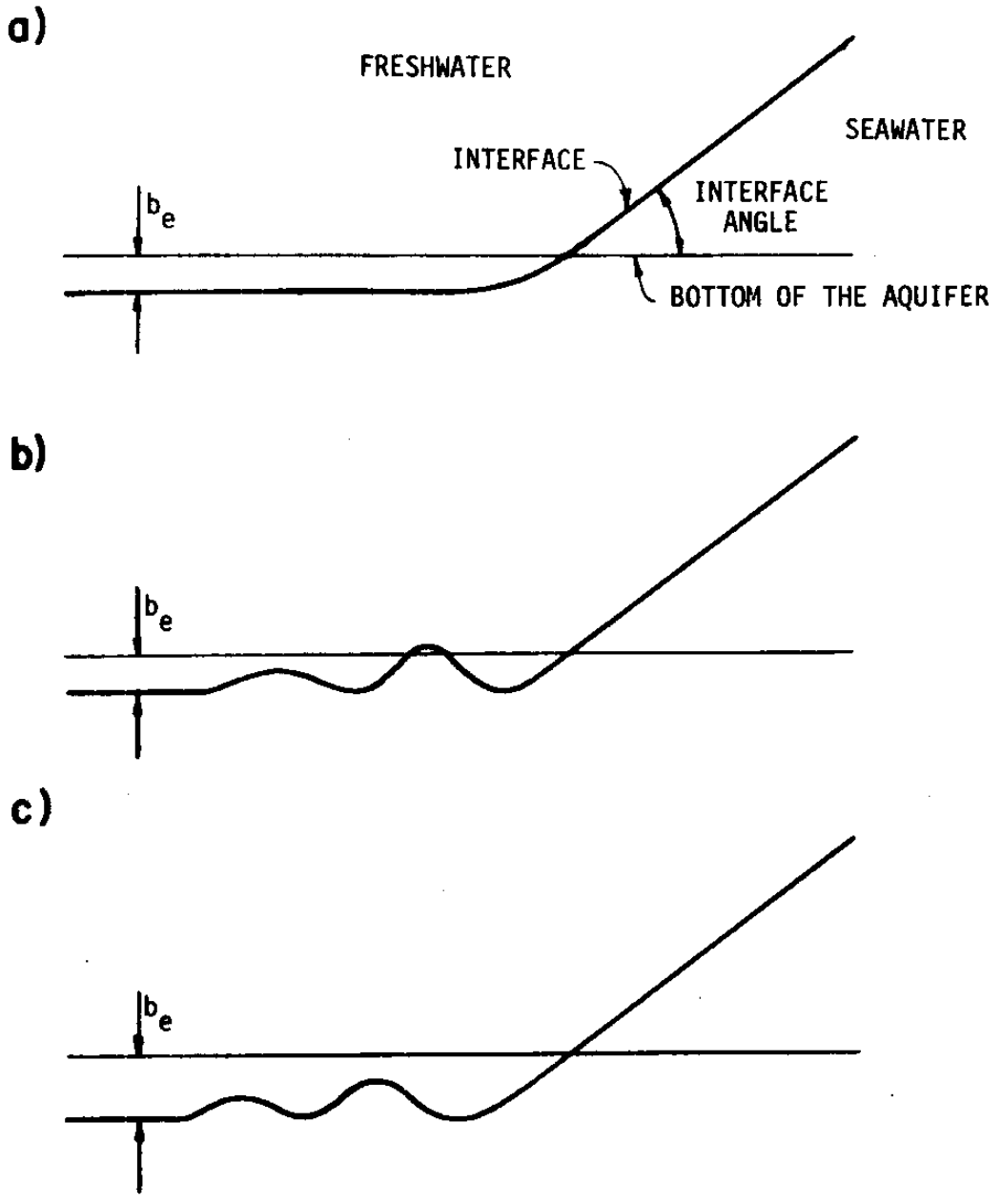


Figure 4.16 Numerical Oscillations of the Interface Inland of the Toe:
a) Ideal Representation; b) Thin Extra-Layer;
c) "Thick" Extra-Layer.

these oscillations seems to be related to the angle between the interface and the bottom of the aquifer: larger angles tend to lead to oscillations. In the gravitational segregation problem this angle is larger than those usually observed in field problems.

4.3.4.2 Sensitivity to the Number of Gauss Points

To study the sensitivity of the toe tracking algorithm to the number of Gauss points, 2 × 2, 3 × 3 and 4 × 4 Gauss points were used to integrate the conductivity matrix within elements containing a toe. The results of these runs using grid 1 of Figure 4.18 were essentially identical; therefore, simulations with a larger grid, grid 5 of Figure 4.18, were performed. This grid uses elements twice as long as those of grid 1 of the same figure. Again the results obtained are indistinguishable when plotted at the scale of Figure 4.17, where only the results for the case of 4 × 4 Gauss points are shown. Table 4.1 presents numerical values of the toe location, L, for the analytical solution and for the three cases of Gauss point density; again there are only minor differences in the numerical solutions. Table 4.2 presents the mean and standard deviation of the differences from the analytical solution for each case. The 2 × 2 solution shows the smallest mean but the largest standard deviation, and the 4 × 4 solution the smallest standard deviation. Despite this minor improvement in solution accuracy, and because of its small marginal cost, 4 × 4 integration is recommended for elements containing the interface toe, particularly for highly distorted elements.

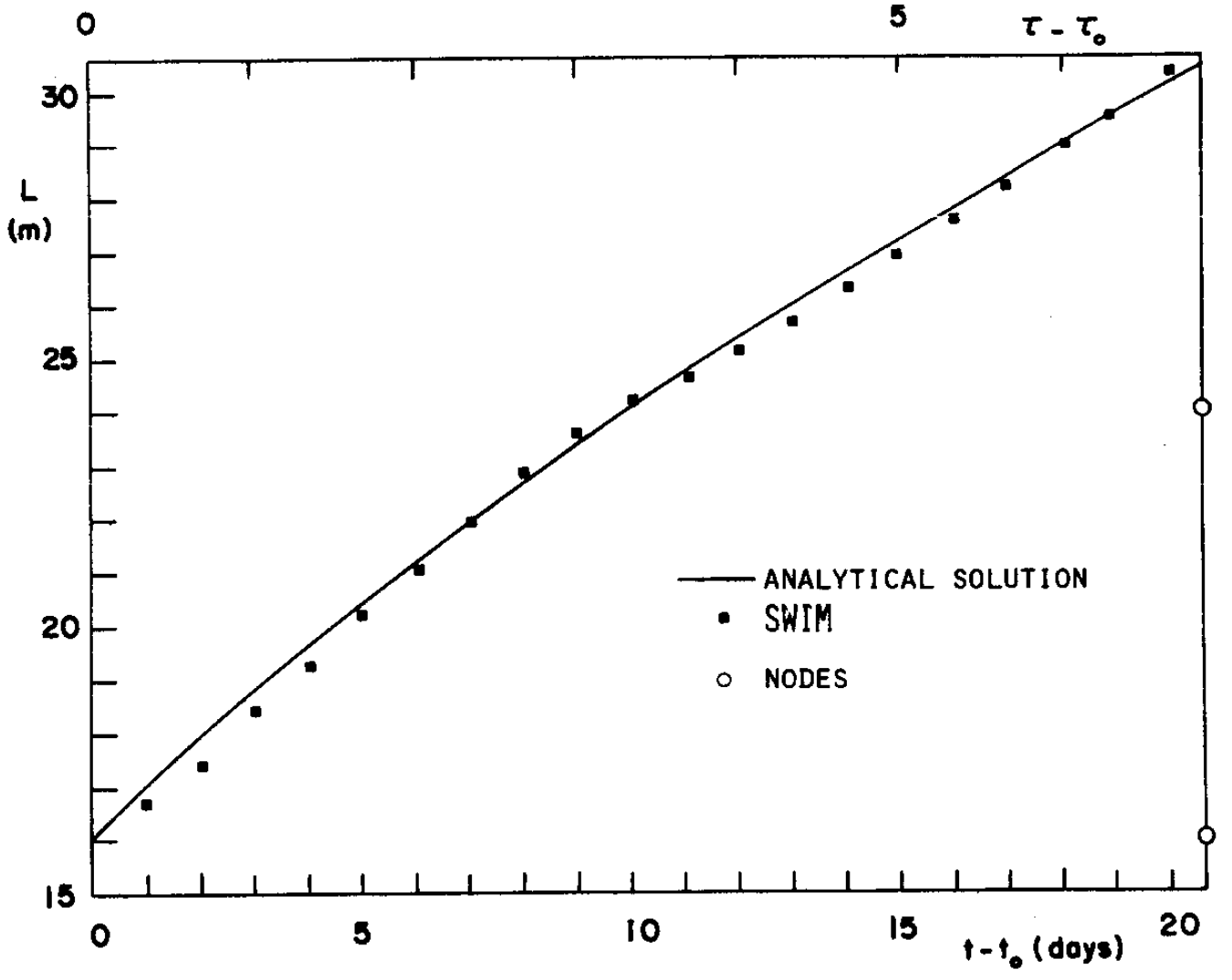


Figure 4.17 - Sensitivity of the Toe Movement to the Number of Gauss Points, Large Linear Grid (Grid 5) and 4×4 Gauss Points

Table 4.1 Toe Location Predictions for Various Densities of Gauss Points Using Grid 5

$t-t_o$ (days)	Toe Location L (m)			
	Analytical	Number of Gauss Points		
		4 × 4	3 × 3	2 × 2
1	16.988	16.663*	16.663*	16.663*
2	17.918	17.432	17.431	17.435*
3	18.803	18.287	18.286	18.296*
4	19.649	19.203	19.202	19.221*
5	20.460	20.146	20.147	20.175*
6	21.240	21.089	21.097	21.158*
7	21.992	21.998*	22.013	22.041
8	22.719	22.854*	22.867	22.891
9	23.424	23.634*	23.645	23.665
10	24.108	24.196	24.188	24.168*
11	24.773	24.643*	24.636	24.616
12	25.421	25.149*	25.141	25.122
13	26.053	25.708*	25.699	25.682
14	26.670	26.315*	26.304	26.291
15	27.273	26.959*	26.946	26.941
16	27.863	27.629*	27.614	27.622
17	28.440	28.311	28.297	28.323*
18	29.006	28.994*	28.981	29.025
19	29.561	29.664	29.655*	29.710
20	30.106	30.315	30.313*	30.372

Note: The * indicates the numerical solution closest to the analytical solution.

Table 4.2

Statistics of the Deviations from the Analytical Solution Using Grid 5

Statistics	Number of Gauss Points			Basic Simulation
	4 × 4	3 × 3	2 × 2	
Mean	-0.164	-0.167	-0.152	-0.028
Standard Deviation	0.232	0.235	0.248	0.123

4.3.4.3 Sensitivity to the Space Discretization

Figure 4.18 presents the finite element grids used for this analysis. Grid 1 was used for the basic simulation; grid 2 is a more detailed 4-node linear grid with four times more elements in the zone of toe movement; grid 3 uses the same number of nodes as grid 1, but with 6-node quadratic elements; grid 4 has the same element size as grid 1, but uses quadratic 6-node elements in the zone of toe movement; grid 5, already described in the previous section, has linear elements twice as long as those of grid 1. A constant time step Δt = 1 day and a constant thickness of the extra-layer of 10^{-3}b was used for the simulations with the five grids. In the case of grid 2, this gives less accurate results, because the toe moves across more than one element in one time step, a condition to be avoided (see Section 4.3.4.4).

All five grids resulted in accurate representations of the interface, but some of the grids did not perform as well in locating the toe.

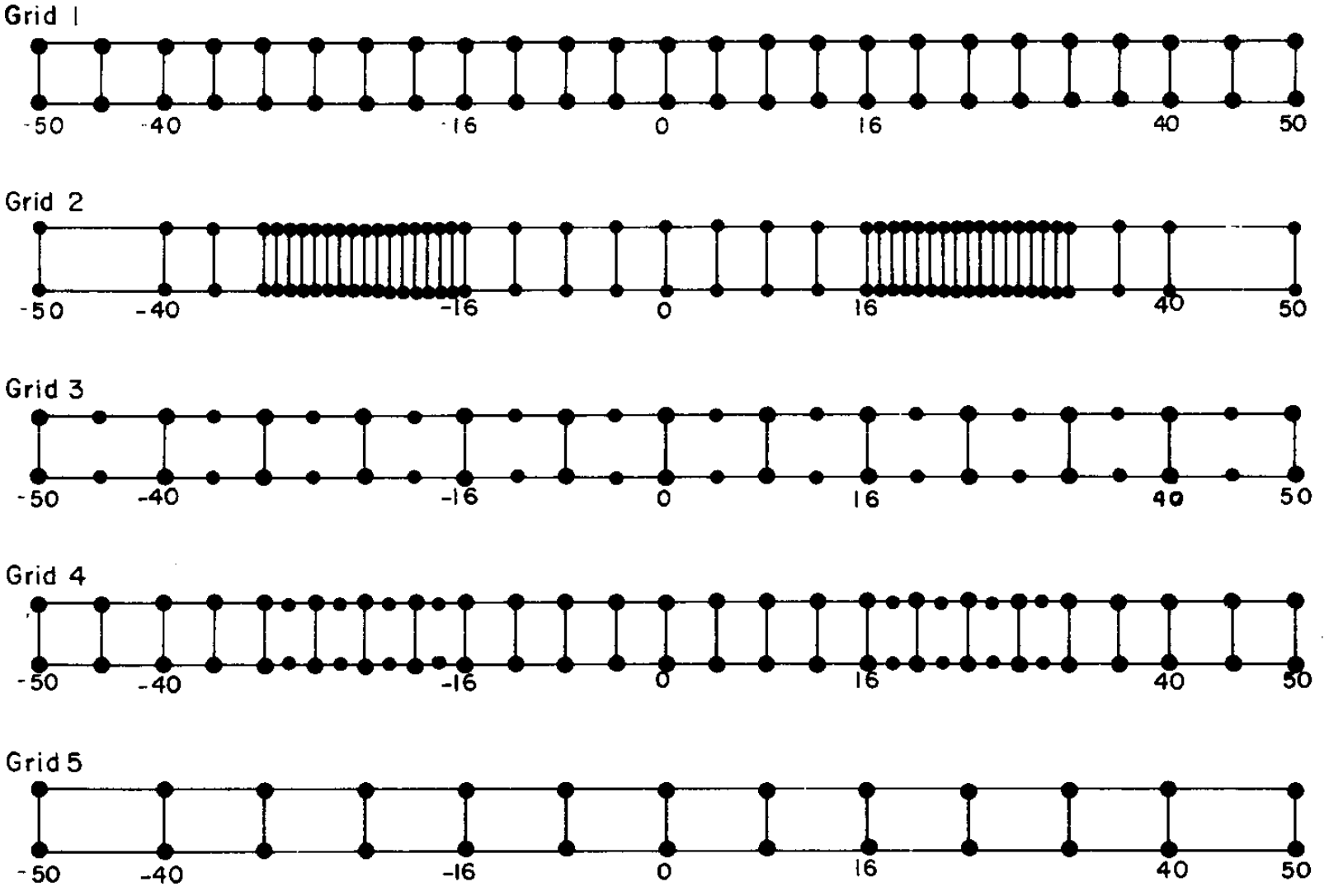


Figure 4.18 - Finite Element Grids Used in the Gravitational Segregation Problem

Figures 4.19 and 4.20 present the toe locations for these simulations; the basic simulation results using grid 1 are plotted in both figures, to be used as a term of comparison.

Figure 4.19 shows the results of the simulations using grids 2 and 4, both containing more nodes than grid 1 in the zone swept by the toe. Additional node density should provide a more accurate toe location. However, because of the extrapolation method of toe location, it was found that it is advisable to have this detail also extend to the element just preceeding the toe, instead of having the initial toe position coincident with the boundary of a significant change of element size and/or type. One of the consequences of not providing this transition zone in the grid is shown in Figure 4.19, in which an initial acceleration of the toe is present. The model could never recover from it. The predicted toe location for grids 2 and 4 is correct in pattern, but shifted. Using grid 2 and grid 4, but a later initial condition, at $t_o' = t_o + 5$ days, additional simulations were performed, and the results are shown in Figure 4.21. In this case the grid is now detailed on both sides of the initial toe location and the toe prediction is significantly improved.

Another reason for the shift of Figure 4.19 is the size of the time step, especially for the case of grid 2. In order to have an accurate representation of the toe location, the toe should take at least one time step to sweep through any element. When this condition is not satisfied, the toe position is accelerated. After the first time step, in the example of grid 2, the theoretical location of the toe is pract-

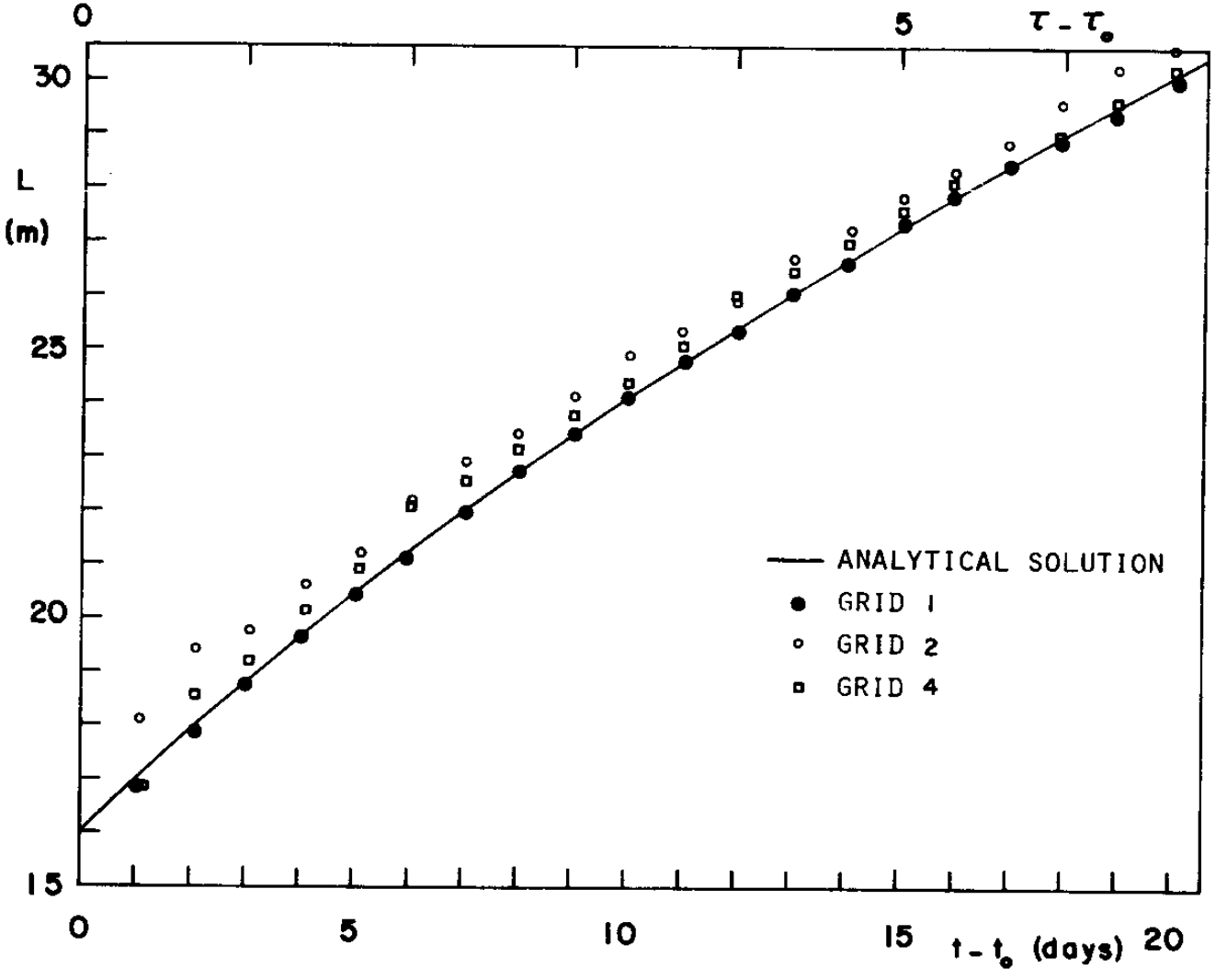


Figure 4.19 - Sensitivity of the Toe Movement to the Space Discretization (1)

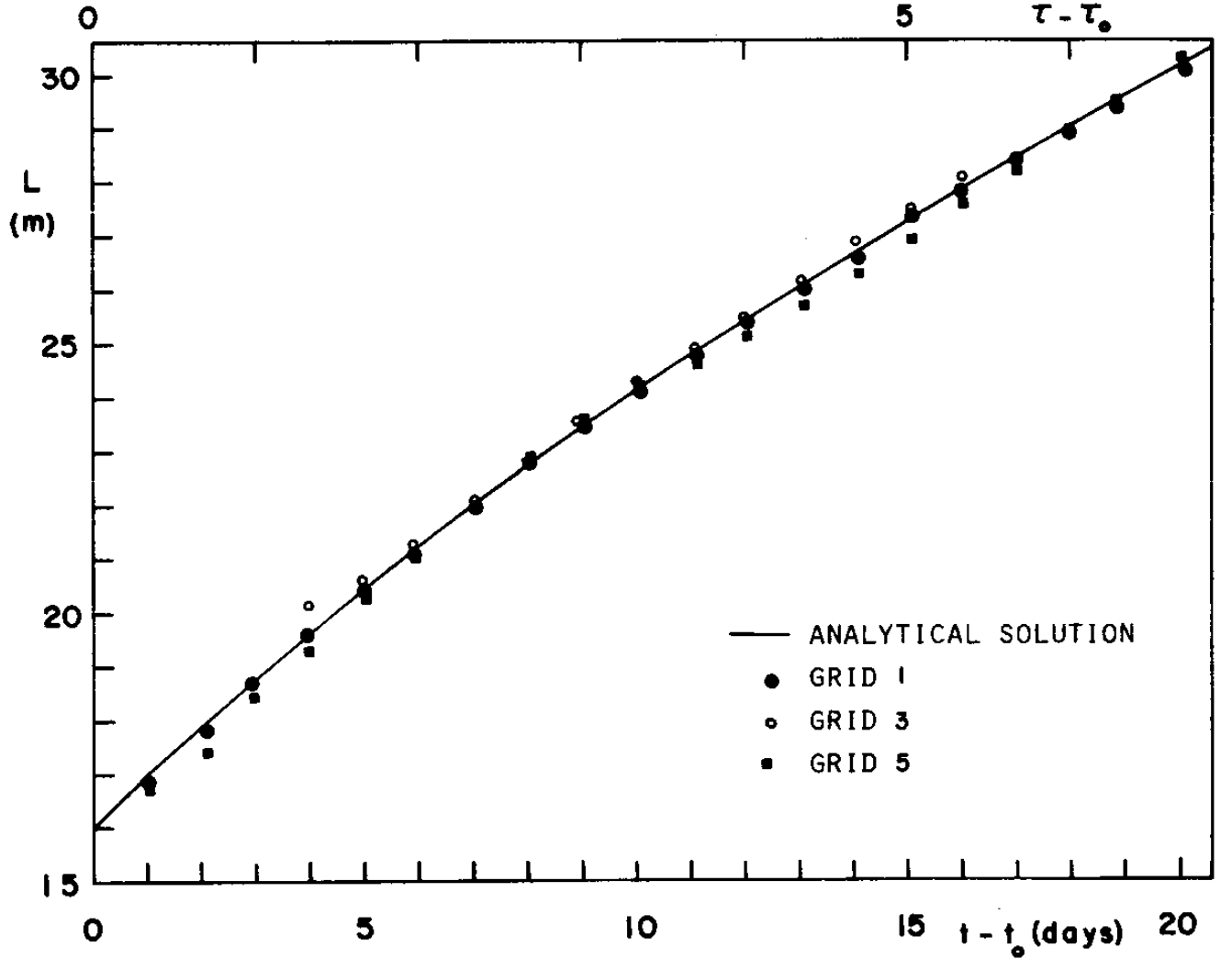


Figure 4.20 - Sensitivity of the Toe Movement to the Space Discretization (2)

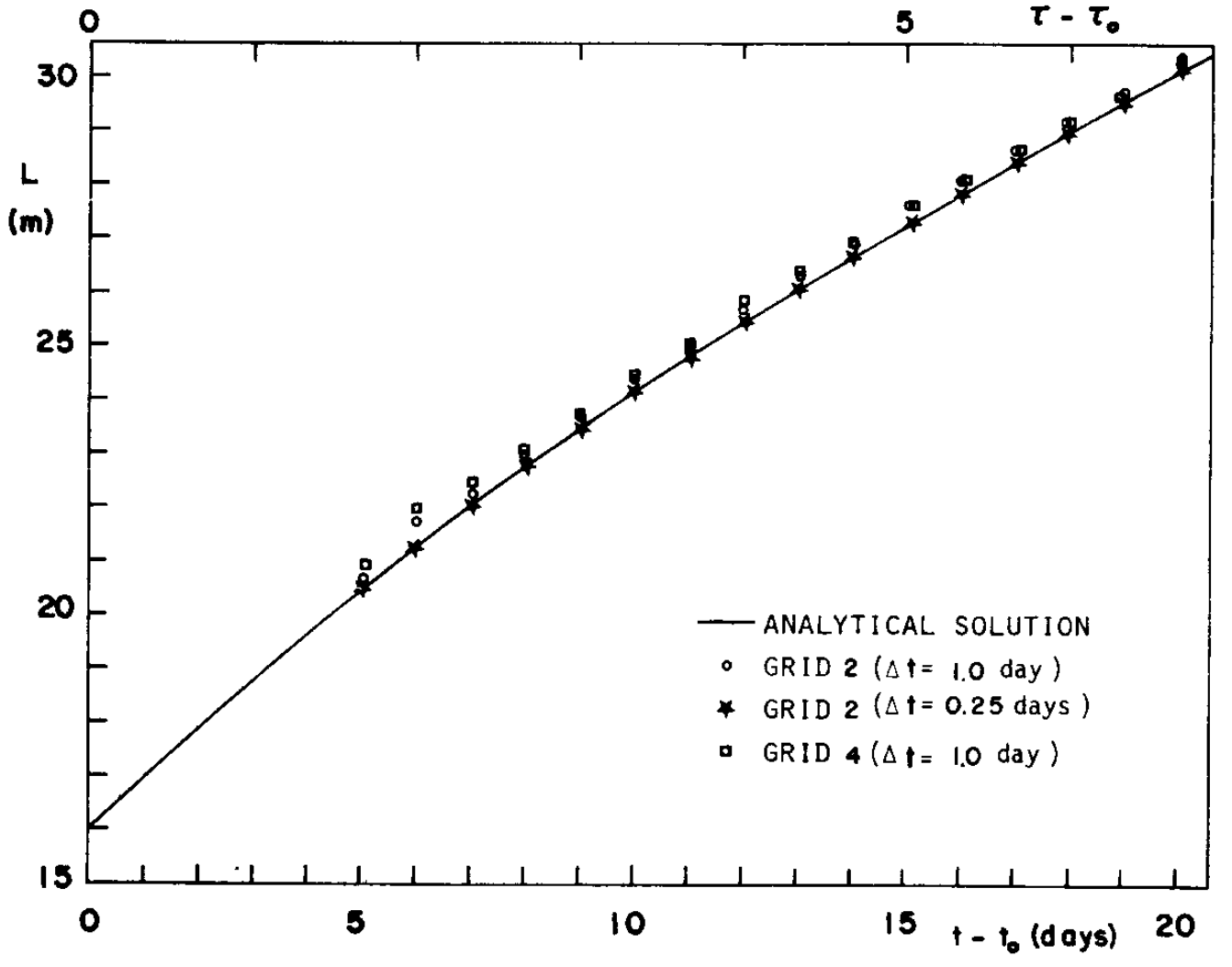


Figure 4.21 - Prediction of the Toe Movement when a Grid Transition Zone is Provided for Grids 2 and 4

ically at the end of the first small element (16.986 m vs. 17.0 m). As a result the model is unable to capture the movement of the toe across the element and a less accurate prediction results. The model locates the toe at 18.05 m in grid 2; that is, at the beginning of the third small element, and over a full meter from its proper location. However, if a smaller time step is used, for example one fourth of the original one, $\Delta t = 0.25$ days, a much better result is obtained, as shown in Figure 4.22. An even better result is shown in Figure 4.21 when a later starting time is used to avoid the element size discontinuity problem.

The results obtained with grids 3 and 5, as shown on Figure 4.20 are satisfactory. Both grids have the same element size, but grid 3 uses 6-node quadratic elements versus the 4-node linear elements of grid 5. The results obtained with grid 3 are better than those obtained with grid 5, as expected. One of the reasons that the results with these two grids are comparable in quality with the results of the basic run is that the toe takes twice as much time to sweep the elements, which are twice as large, than is the case of the basic simulation. The results are also not as good as the basic simulation because the elements are twice as big, and therefore, there are only half of the number of Gauss points for the same area.

In summary the quadratic elements perform better than linear elements of the same size, but at a higher cost due to the increased bandwith. The primary advantages of the quadratic elements are a better representation of the boundaries and of the interface and piezometric head surfaces. However, in this particular problem these advantages are

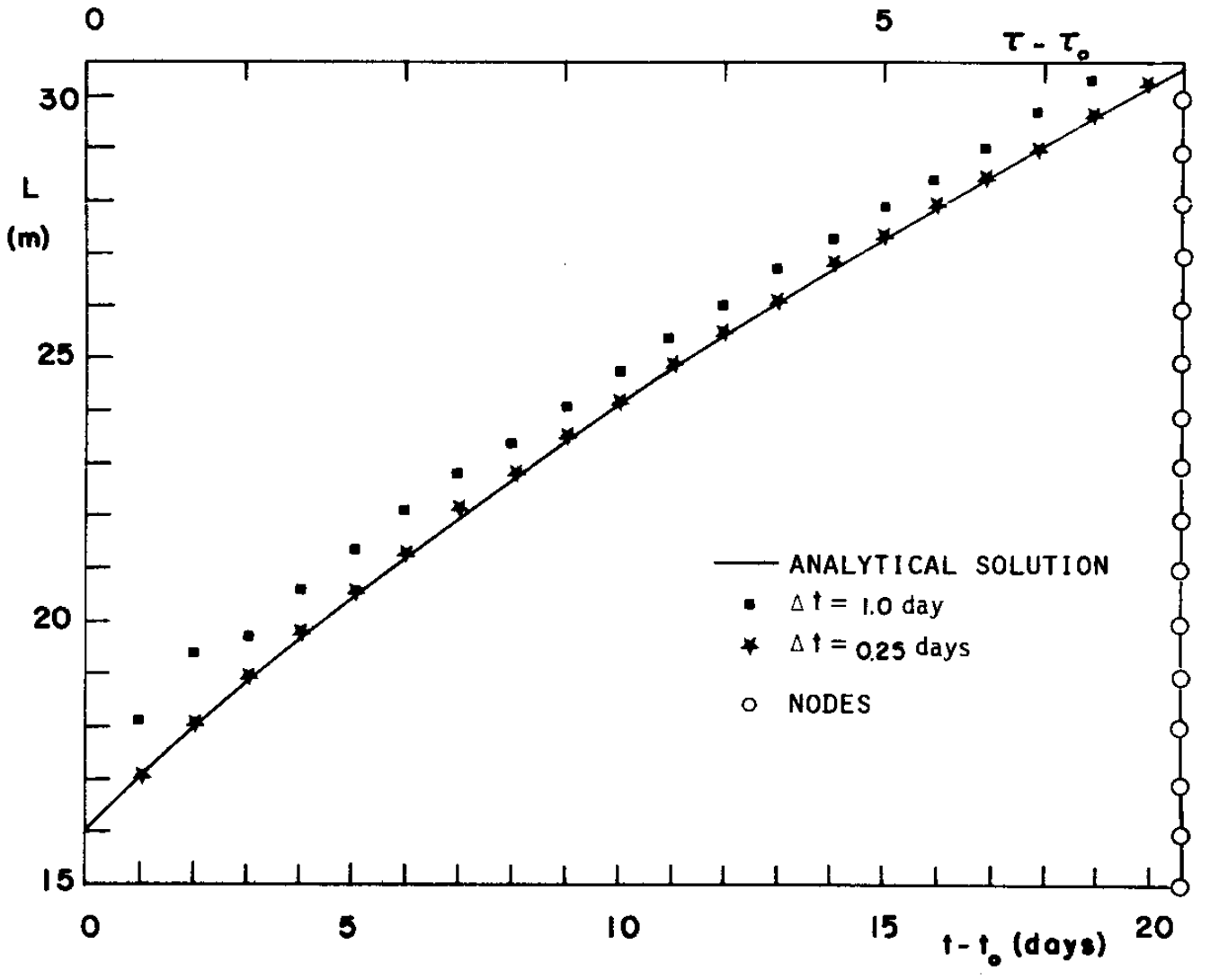


Figure 4.22 - Sensitivity of the Toe Movement to the Time Step Size when Grid 2 is used

not so apparent, because the interface is a plane, and linear elements represent it precisely.

A comparison of costs for these simulations is of interest. The computer used was an IBM 370/168 with a virtual memory system (VM/CMS). The CPU times are compared in Table 4.3 in terms of ratios relative to the basic simulation which took ≃ 32 CPU seconds.

The accuracy of a problem solution depends on the selection of grid detail and time step size, both of which also determine the cost of the run. Some guidelines for the selection of space and time discretization are summarized at the end of this chapter.

4.3.4.4 Sensitivity to the Time Step

The selection of time step depends on a tradeoff between accuracy and cost, with the extra-layer thickness playing a role that has not

Table 4.3

Comparison of CPU Times Used with Different Grids

Grid	Ratio of		Type of Grid	Half Band Width	
	CPU Time	Number of Nodes		Mean	Maximum
1	1	1	linear	7	8
2(Δt=1 day)	1.60	1.96	linear	7	8
3	1.06	1	quadratic	9	12
4	1.21	1.32	linear/quadratic	8	12
5	0.54	0.52	linear	7	8
2(Δt=1/4 day)	5.43	1.96	linear	7	8

yet been clarified. As a general rule the smaller the time step, the better the accuracy and higher the cost. However, for a given space discretization, there is a value of the time step below which there is no improvement of solution accuracy for given boundary conditions. There is also an upper limit of the time step dictated by the physics of the phenomenon. If the problem is cyclic this value roughly ranges between 1/20 and 1/40 of the period; if the problem is not repetitive then the upper limit of the time step is imposed by the temporal variability of the boundary conditions and the total solution time.

These are some of the broad issues behind the time step sizing problem. The restriction on the time step imposed by the toe movement are superimposed on these; the toe movement restriction is usually more binding. Since the toe movement is tracked by the Gauss points, a very rapid movement through several Gauss points in one time step is an undesirable feature, leading to local acceleration of the solution, as already shown in Figures 4.19 and 4.22 for the grid 2 simulation. A reasonable upper limit for the time step is that the toe should not move across an entire element in only one time step. A lower limit, below which no significant improvement on the accuracy of tracking of the toe is expected, is half the time it takes the toe to travel between the two closest Gauss points in the grid.

From these principles one can infer that the smaller the grid in the vicinity of the toe, or the more Gauss points used, the smaller the required time step. The celerity of the toe is largest at the initial time step, $\tau = \tau_o$ or $t = t_o$, for the example problem. From Eq. 4.2:

$$v\Big|_{\tau_o} = \frac{\partial(L/b)}{\partial\tau}\Big|_{\tau_o} = \frac{1}{2}\frac{1}{\sqrt{\tau}}\Big|_{\tau_o} \simeq 0.31 \qquad (4.3)$$

Applied to grid 1 of Figure 4.17, with 4 × 4 Gauss points, the following limits for the time step are found:

Lower limit

- length of the side $\Delta x = 4$ m
- smallest distance between Gauss points $\Delta x \times 0.14 = 0.56$ m

$$\Delta\tau_{min} = (\frac{0.14}{2}\frac{\Delta x}{b})/v \simeq 0.09 \simeq 1/11$$

or

$$\Delta t_{min} \simeq 3/11 \simeq 0.27 \text{ days}$$

Upper limit

$$\Delta\tau_{max} = (\frac{\Delta x}{b})/v \simeq 1.28$$

or

$$\Delta t_{max} \simeq 3.84 \simeq 4 \text{ days}$$

For the standard simulation the time step used is approximately four times the minimum, $\Delta\tau = 1/3$ or $\Delta t = 1$ day.

Simulations using a time step of $\Delta\tau = 1/12$, 1/6, 2/3 and 4/3, or $\Delta t = 1/4$, 1/2, 2 and 4 days, were performed and the results are shown in Figure 4.23. As expected the larger time steps result in less accurate predictions of the toe location; the interface position is well modeled, however. The smaller time steps slightly under predict the location of the toe, which is explained by the relationship between

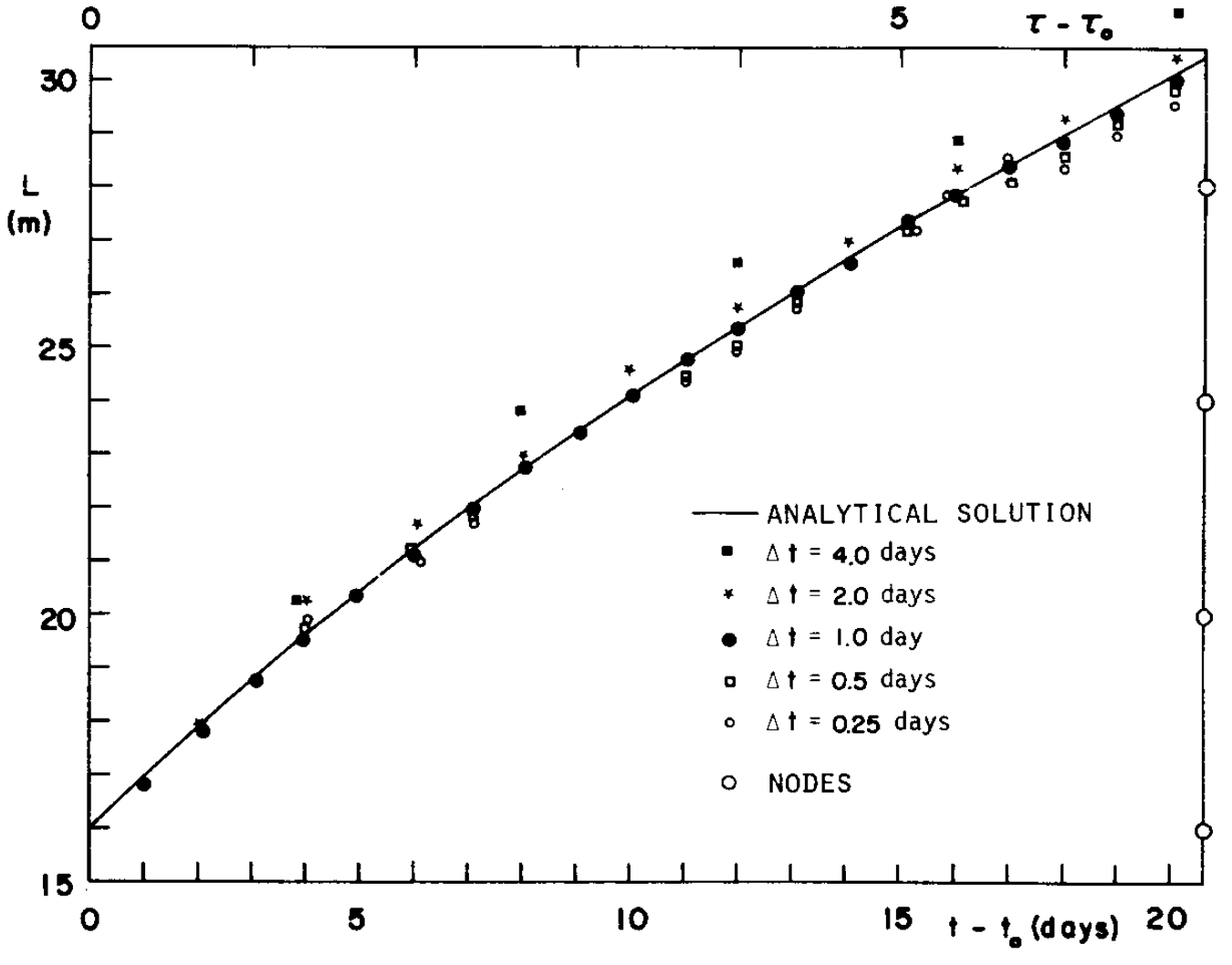


Figure 4.23 - Sensitivity of the Toe Movement to the Time Step
(Solution Shown Only for Integer Values of Time)

time step and extra-layer thickness pointed out in Section 4.3.4.1. Looking back at Figure 4.15 one can see that a smaller time step performs better with a smaller extra-layer thickness. This fact increases the complexity of the space and time discretization selection process, because the selection of the extra-layer thickness also plays a role. Generally, one selects the grid and the time step first, and afterwards selects the extra-layer thickness.

Table 4.4 presents the CPU time required for this different simulations. An interesting fact shown in this table is that with the exception of the smaller time step, the CPU ratio increases geometrically with decreasing time step, at a rate constant of 1.53. Perhaps the reason for the exception of the smaller time step, a CPU ratio of 2.95 instead of 2.37, is due to the additional effort in performing marginal tasks in the code.

Table 4.4

CPU Time for Simulations with Different Time Steps

$\Delta\tau$	Δt (days)	Ratio of	
		CPU Time	Total Number of Time Steps
1/12	0.25	2.95	4.0
1/6	0.5	1.55	2.0
1/3	1	1.0	1.0
2/3	2	0.65	0.5
4/3	4	0.43	0.25

4.3.4.5 Sensitivity to the Number of Time Steps Between Reformation of the Conductivity Matrix

In the simulations described thus far the effective conductivity matrix, K* (defined in Table 3.1) has been updated every time step, which is the best way to keep track of the changing interface in SWIM. However, if the interface does not move rapidly there is no need to update this matrix every time step, saving some computer time. Simulations have been performed in which the effective conductivity matrix is updated, respectively every 2, 4 and 8 time steps. As demonstrated in Figure 4.24 the results for the first two simulations are very close to those for the basic simulation, with updating every time step. The results of the simulation with updating every eighth time step are not as good but still acceptable. For the time step $t-t_o$ = 13 days the out of balance flows were larger than the incremental flows after 10 iterations; in this situation the code automatically updates the effective conductivity matrix, to insure solution convergence. A simulation with updating specified every tenth time step was also attempted. But, again, the code automatically updated at the 9th and later at the 16th time step anyway, to obtain convergence.

The CPU time requirements for these simulations are presented in Table 4.5, once again in terms of the CPU time required in the basic simulation. The savings obtained when going from every 2 to every 4 time steps is marginal. Updating every 8th time step takes more CPU time, because of the number of iterations used when no convergence was obtained, at $t-t_o$ = 13 days.

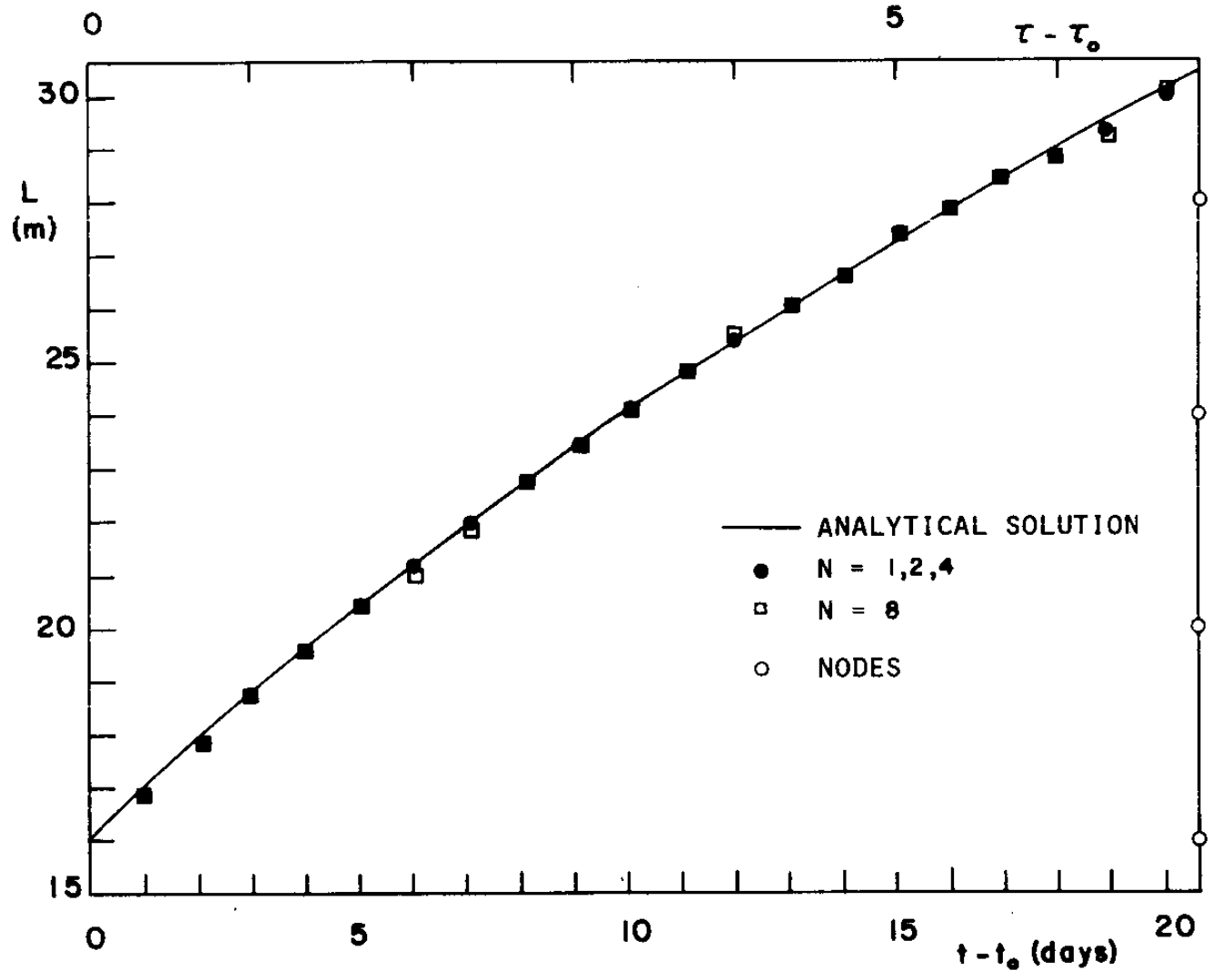


Figure 4.24 - Sensitivity of the Toe Movement to the Number of Time Steps (N) Between Updating the Conductivity Matrix

Table 4.5

Comparison of CPU Time Used with Different Reformation Intervals for the Effective Conductivity Matrix

Number of time steps between reformation	Ratio of CPU time
1	1
2	0.89
4	0.87
8	1.08

4.3.4.6 Sensitivity to the Consistent Versus Lumped Storage Matrix

All the results presented thus far have used a consistent storage matrix. It is generally known that a lumped storage matrix will not perform as well for the type of boundary non-linearity modeled by SWIM. Nevertheless simulations using a lumped matrix and grid 1 of Figure 4.18 were run. Different values for the extra-layer thickness were used to judge from the influence of this factor with a lumped storage matrix. The resulting toe locations are shown in Figure 4.25; they are the least accurate reported in this chapter. However, it should be noted that the interface itself is reproduced far better than these toe location estimates may suggest. This will be demonstrated in Section 4.4.

No difference in CPU time was noticed for the two runs. Therefore, as a general rule the lumped storage matrix should not be used. It is

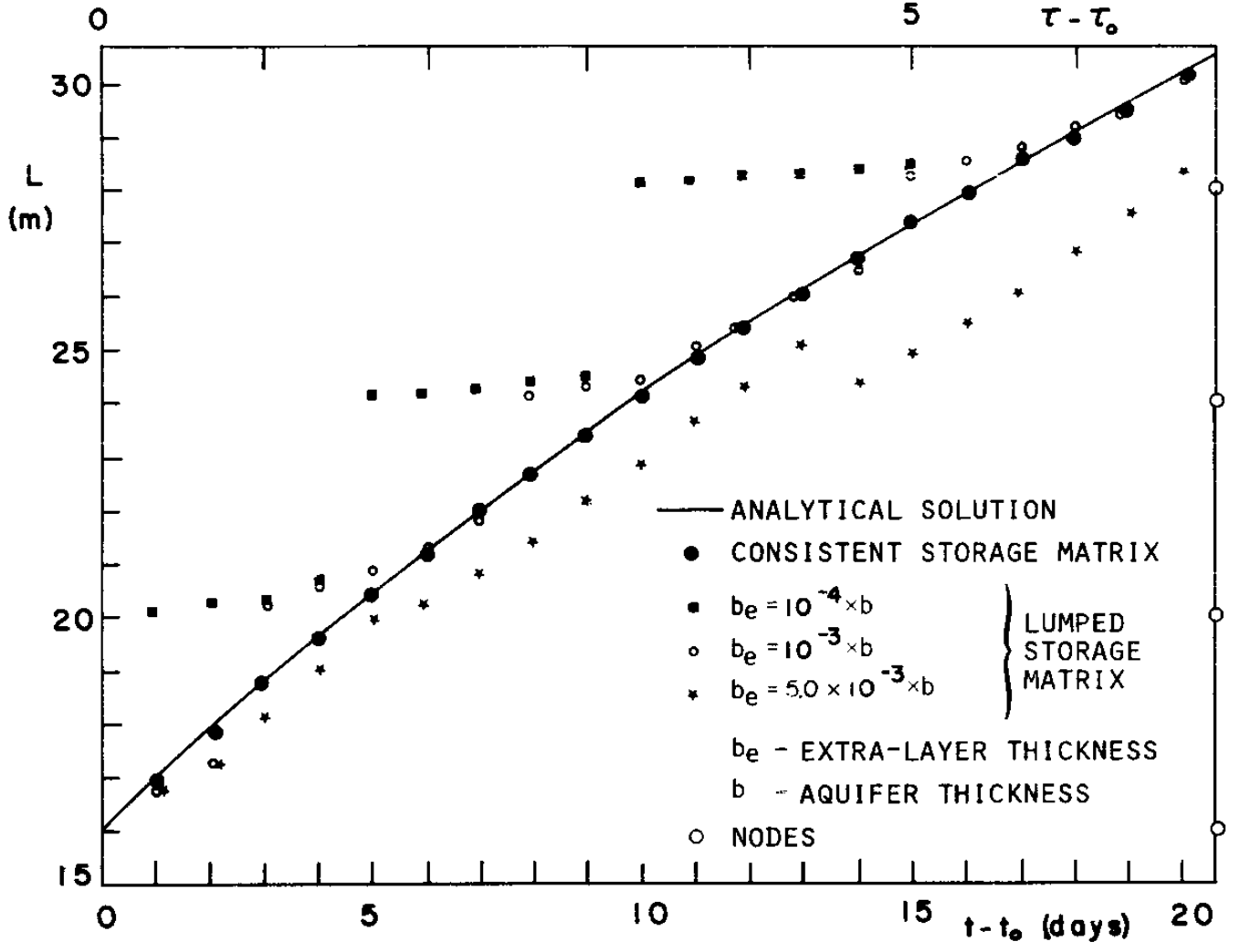


Figure 4.25 - Sensitivity of the Toe Movement to the Use of Consistent or Lumped Storage Matrix

expected to perform even less satisfactorily when distorted quadratic elements are employed. Alas, there is an exception. Lumping the storage matrix improves the positive-definiteness of the effective conductivity matrix, K*. In certain situations of time step, grid geometry and physical properties, lumping may be required in order to avoid a non-positive semi-definite matrix and to solve the system of simultaneous equations.

4.3.5 Post-Processing Calculation of the Toe Location

The toe position is calculated only for output purposes. A post-processing calculation of the toe location makes it easier to distinguish the real toe from fictitious ones that sometimes are produced by oscillations of the interface inland of the lower toe (see Figure 4.16b).

SWIM was designed to be used for the simulation of field problems that often require irregular finite element grids. Therefore the location of the toe, for output purposes, should be obtainable for any type of grid that may be encountered. The most obvious location technique is the use of *interpolation* functions at the elements containing a toe or front. These functions can be used to determine, for example, the intersection of the front with the side of the element. This has the advantage of being relatively simple to implement because it reduces a 2-D problem to a 1-D problem along element sides. However, considering the case of the linear element of Figure 4.26a,b, the interface position at two nodes are required: one at the node immediately inland of the lower toe (node A), and the other at the node immediately seaward of the same toe (node B). Due to the way toe

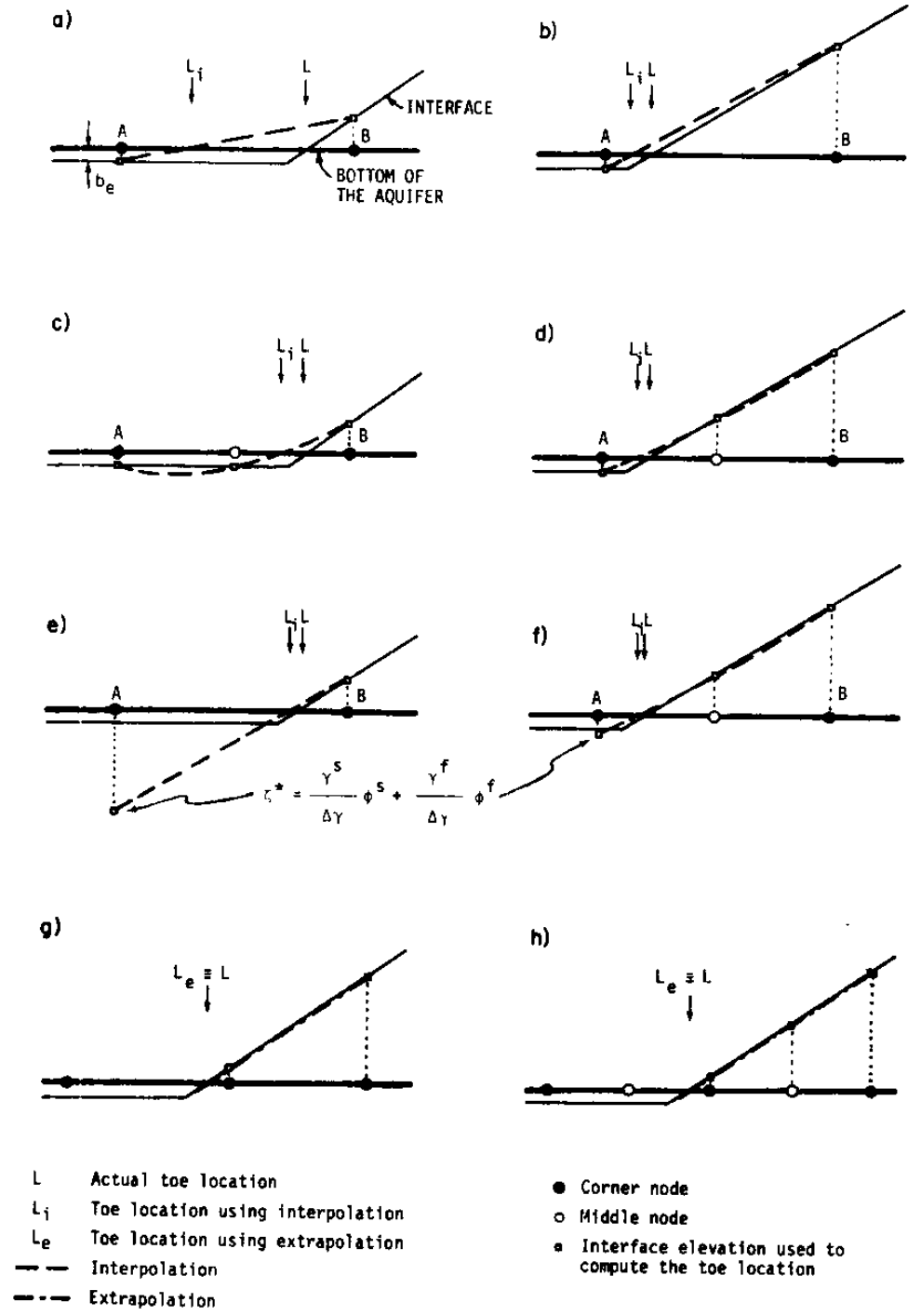


Figure 4.26 - Post-Processing Calculation of the Toe Location: Interpolation: a), b), c) and d); Interpolation Ignoring the Bottom: e) and f); Extrapolation: g) and h).

tracking algorithm works, the "interface" depth below the bottom of the aquifer at node A is very small compared with the interface elevation above the bottom of the aquifer at node B. With a linear interpolation the toe will always be located at the end of the element near node A, as shown in Figure 4.26a and b (for the case of a quadratic element, see Figure 4.26c and d). Instead of using the actual "interface" elevation at node A, another surrogate interface elevation can be calculated using the piezometric heads at node A and the interfacial boundary condition (Eq. 3.3) by ignoring the presence of the aquifer bottom and the artificial limit to the depth of the interface below it. This new value is only used for output purposes and produces more accurate results, as shown in Figure 4.26e, f. As a matter of fact, this is identical to the method used for checking whether or not a Gauss point is inland or seaward of the toe. This method can be expensive, depending on the complexity and distortion of the grid.

Another method, that is cheaper and easier to implement, uses *extrapolation* along the element sides. Extrapolation fits a straight line through the interface deviation at the two nodes seaward of the toe in the case of linear extrapolation, or a parabola through the three seaward nodes in the case of quadratic extrapolation, regardless of whether or not these nodes are corner or middle nodes (see Figure 4.26g and h). Extrapolation is easy to implement if the grid is regular; that is, if the sides along which the toe is located define a straight line. Otherwise the computational effort involved is significant.

The location of the toe for output purposes is definitely problem dependent, usually grid dependent, and is a question that should be tailored to the requirements of the user. For these reasons it was decided to handle this question using post-processing.

Both linear and parabolic extrapolations were used. The linear extrapolation demonstrated a slightly better agreement with analytical solution, which is not surprising because the shape of the interface is a straight line. However, in other problems in this work (see Chapter 5) linear extrapolation was also used for both linear and quadratic elements because of: its simplicity, ease of implementation for complicated grids and its accuracy. In one simulation, described in Section 5.7, the interpolation functions were used, with excellent results. As explained above, the interface location inland of the toe was computed, in this example, using the piezometric heads.

As a final note on this subject, it is suggested that the location of the toe for output purposes can also be very satisfactorily achieved by using graphic packages that are currently available for plotting contour lines and perspectives of surfaces. This was not attempted in the present work.

4.4 Summary and Guidelines to the Use of Gauss Points to Track the Toe Location

The previous section of this chapter has compared the different solution options in terms of the model's ability to track the toe location. Only brief comments have been added suggesting that the overall shape of the interface has been modeled more or less accurately, with

the only exception concerning the nodes located near the toe. To let the reader judge this accuracy, Figure 4.27 presents the "worst" five simulations in terms of toe tracking presented in this chapter, together with the basic simulation. Thus, in Figure 4.26a, the following simulations are presented: from Figure 4.14, an extra-layer thickness of 10^{-2} b; from Figure 4.23, Δt = 2 days and Δt = 4 days. Figure 4.27b presents the results of the basic run and two runs, from Figure 4.25, using lumped storage matrix with extra-layer thickness of 10^{-4}b and $5x10^{-3}$b. In all of these "worst" cases, the model reasonably predicts the interfaces location, except near the toe. Further comparisons with Shamir and Dagan's (1971) mesh regeneration solutions are shown in the next chapter.

It has been demonstrated by the results of the different simulations that the solution accuracy depends on a relationship between element size, time step and extra-layer thickness, and that these are the three more important factors to be considered when preparing the data. Other considerations, not as important as these three, are: permeability of the extra-layer, fixed at 10^{-5} times the permeability of the aquifer; use of a consistent storage matrix; and the number of time steps between updating the effective conductivity matrix, K*. Updating K* every time step always works well, but is not always necessary, especially for a slowly changing interface and/or simulations with small time steps. When K* is updated periodically it is advisable to set the number of time steps between reforming K*, to be a

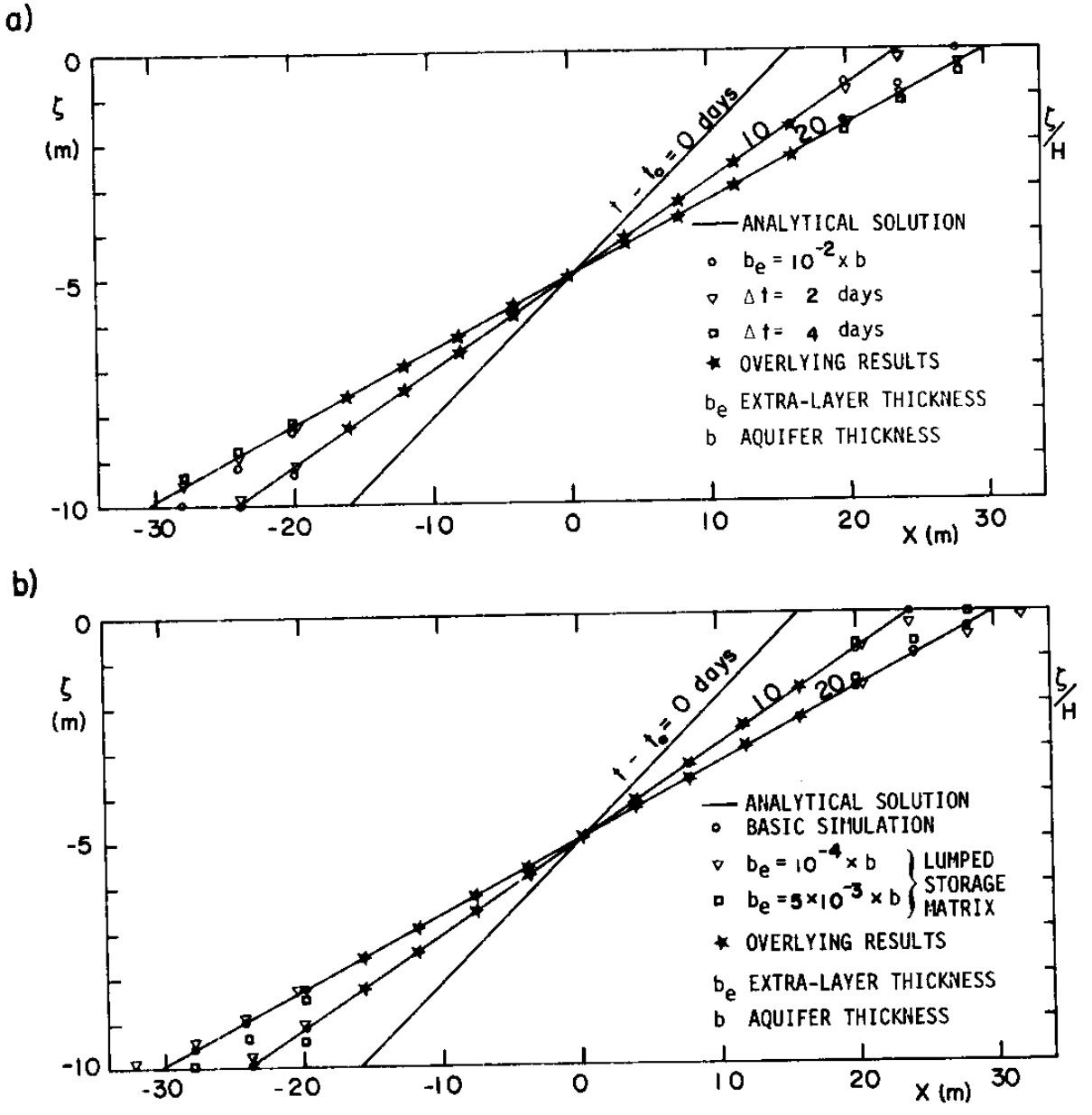


Figure 4.27 - Interface Location for Various Parameters and Conditions

submultiple of the number of time steps between outputs. That is, K* should always be updated when an output is requested, in order to insure the best possible recorded solution.

Associated with the grid and time step sizes are the number of Gauss points (order of quadrature) used for conductivity matrix integration in those elements containing a toe. The order of quadrature for these elements should be at least the same as that used for elements not containing a toe. Since the number of elements with a toe is usually small compared with the total number of elements in a grid, a 4×4 quadrature is recommended because additional CPU time is marginal and it will insure a less variable solution for any type of element.

The required degree of solution accuracy for toe tracking determines the size of the elements. As a rule of thumb, a maximum error of the order 1/4 of the element size, in the direction of the toe movement, can be expected, and this will determine the size of the grid. The best type of element depends on the shape of the interface. Linear elements have performed very well in these tests and they show some savings on CPU time due to a smaller bandwidth, but one should remember that in these examples the interface is linear. Theoretically and practically, quadratic elements perform better. But the increase in accuracy does not always justify the increase in cost. The choice of the type and size of element, as with many other elements of numerical modeling is a question of experience. Only with time can one learn to make the best decision at first try.

The selection of the time step should be based on the variability

of the boundary conditions, and the condition that the toe should take more than one time step to move across any element it encounters. To be more specific the toe should travel the distance between the two closest Gauss points no slower than two time steps and no faster than half a time step. The distance between Gauss points is given in Table 4.6 in terms of the length of the shortest element side, ℓ. Note that the shortest element side should be oriented parallel to the direction of the toe movement.

Based on the experience in this work the extra-layer thickness is best given by $b_e = 10^{-3}b$, where b is the thickness of the aquifer. It is combined with a time step that allows the toe to cover the distance between the two closest Gauss points in one time step, in a 4 × 4 scheme. Under these constraints $b_e = 10^{-3}b$ has provided the best results and is recommended in general. However, if one selects a smaller time step a smaller extra-layer thickness should be specified, for example $10^{-4}b$ for the minimum time step recommended above.

Table 4.6

Distance Between Gauss Points

Order of Gauss Quadrature	Smaller Distance Between Gauss Points
2 × 2	0.42 × ℓ
3 × 3	0.22 × ℓ
4 × 4	0.14 × ℓ

In this chapter a method has been presented that indirectly tracks the toe. It has been shown to accurately represent the toe position as a function of time, as well as the interface itself. Sensitivity analysis has been performed to identify the important parameters in the algorithm. They prove to be: grid type and size and time step size, as in any finite element model; and thickness of the extra-layer, b_e, a new parameter associated with the Gauss point method of toe tracking.

Chapter 5

VERIFICATION AND APPLICATIONS

5.1 Introduction

Model verification is a necessary step in the development of any type of model, in order to establish its validity and accuracy for describing the phenomena it tries to represent. With numerical models this verification consists of two stages: verification of the conceptual model (the mathematical statement consisting of governing equations, etc.) which is the foundation of the whole approach, and verification of the numerical techniques used to solve the conceptual model (the time and space discretizations, iterative procedures, equation solvers, special algorithms, etc.).

The SWIM code is based on the fundamental assumption that seawater intrusion can be described by vertically averaged equations resulting in a layer of freshwater and an underlying layer of seawater, with an immiscible interface separating them. There are two approximations implied by this approach: a relatively narrow transition zone between freshwater and seawater, and essentially horizontal flow (the Dupuit assumption). There are clearly cases in which one or both of these approximations are violated. For example, the Long Island Magothy aquifer apparently has significant vertical flow disallowing the use of the Dupuit approach to model its seawater wedge (Collins and Gelhar, 1971, 1977). The Biscayne aquifer, south of Miami, Florida, is well recognized for having a significant mixing or transition zone, leading

many researchers to conclude that an immiscible interface approximation for this aquifer is invalid (Lee and Cheng, 1974; Henry, 1964a; Segol and Pinder, 1976). However, there is sufficient basis to judge the immiscible interface/Dupuit approach valid for many applied problems. It is a major thrust of much of the analysis used world around to evaluate seawater intrusion (see Chapter 2 and Bear, 1960; Bear and Dagan, 1963, 1964a, 1964b; Collins and Gelhar, 1971, 1977; Fetter, 1972; Rumer and Harleman, 1963; Hashish et al., 1979; Mualen and Bear, 1974; Pinder and Page, 1976; Shamir and Dagan, 1971; Todd and Huisman, 1959; Van Dam, 1976; Verou, 1978). This report makes the assumption that the immiscible interface/Dupuit model is appropriate. The next question concerns how accurate is the SWIM solution of it?

Chapter 4 presented the SWIM solution of the one-dimensional gravity segregation problem and described a sensitivity analysis of that solution to various numerical considerations: time step, space discretization, space integration, toe tracking algorithm, and its parameters, iteration parameters (number of time steps between reforming of the effective conductivity matrix), and form of the storage matrix. Clearly, the proposed numerical solution performed accurately in the tests documented in that Chapter. However, the computer code has not yet been tested for seawater intrusion problems involving advancing or retreating seawater wedges, or the development of a freshwater lens over seawater, or the presence of leakage flows to a vertically adjacent aquifer, or the injection of freshwater into a two-dimensional aquifer, or the withdrawal of freshwater near a seawater wedge. It is the pur-

pose of this chapter to present SWIM simulations of each of these situations, in order to verify for code's accuracy and versatility. Before this, however, the one dimensional gravity segregation solution of Chapter 4, obtained using SWIM's indirect method of toe tracking, is compared to Shamir and Dagan's (1971) numerical mesh regeneration solution of the same problem.

5.2 Gravitational Segregation: Comparison with the Mesh Regeneration Solution of Shamir and Dagan (1971)

Assuming a sharp interface and the Dupuit approximation, Shamir and Dagan (1971) developed a one dimensional model of seawater intrusion based on finite differences. They used a mesh regeneration scheme with a fixed number of nodes in the interface zone, between the wedge toe and the coastline. The time step was variable, and recalculated each time step, such that the toe movement during that time step was less than the distance between two consecutive nodes. They determined the new position of the toe by linearly projecting, or extrapolating, the interface position, using the values of the interface elevation at the first two nodes seaward of the toe. Shamir and Dagan (1971) initially applied their model to the gravitational segregation problem described in the previous chapter; then they used it to simulate the classical one dimensional seawater intrusion problem. The latter is discussed in Section 5.3.

The gravitational segregation problem presented by these authors is identical to that presented in Chapter 4, with the exception of a slightly later initial condition. They started at t_o = 12.28 days ($\tau_o \simeq 4.1$) instead of t_o = 7.87 days. This corresponds to an initial toe position of L_{t_o} = 20 m instead of L_{t_o} = 16 m, and means that the interface is initially less steep and the toe slower moving than in the case of Chapter 4. To compare SWIM's solution with Shamir and Dagan's result, a new simulation was performed using the same initial conditions (t_o = 12.28 days), grid 1 of Figure 4.18, Δx = 4 m, and all the other conditions of the basic simulation described in Section 4.3.3. This new simulation covered a period of 20 days with a time step Δt = 1 day; that is, 20 time steps were used. In Shamir and Dagan's simulation the nodal spacing varied in time from 2 to 3.2 m in the interface zone, and from 3.2 to 2.7 m elsewhere. A total of at least 19 variable time steps were used, assuming that all the time steps used are plotted in their Figure 8.

An important difference between the two simulations is that the SWIM simulation represented the entire spatial domain of the problem (from -50 m to +50 m), including both the upper and lower toes, while Shamir and Dagan invoked symmetry and modeled only the half space (from -50 m to the origin) and the lower toe. As a consequence the boundary conditions were different in the two simulations; SWIM used a specified head for both freshwater and seawater at only one end of the grid, while Shamir and Dagan used a constant specified interface depth at x = 0 equal to half of the aquifer's thickness, b/2.

In summary SWIM's simulation used a coarser grid with roughly 1/2 to 3/4 the detail, about the same sized time step, and less constraining boundary conditions than Shamir and Dagan's (1971) simulation.

Figure 5.1 presents the comparison between the two simulations for both interface position and toe location. For the interface position, Figure 5.1a, the results are equivalent. For the toe location, Figure 5.1b, Shamir and Dagan's solution presents a slightly better representation at some points, but overall the two solutions are equivalent, especially when SWIM's larger space discretization is accounted for. Clearly, for this example, the indirect technique of toe tracking works as well as the direct method of mesh regeneration.

5.3 Classical One-Dimensional Seawater Intrusion Problem

In this section the solution for the classical Dupuit type one-dimensional seawater intrusion problems in a confined aquifer is presented. The aquifer is assumed homogeneous and istropic, with a constant thickness and a horizontal bottom, and a constant freshwater discharge to the sea. At time $t = t_o = 0$ the discharge is changed from that of the initial equilibrium, and the movement of the seawater wedge observed. Two situations were modeled:

1. sudden reduction of the freshwater discharge to the sea;
2. sudden increase of the freshwater discharge to the sea.

These were two of the situations modeled by Bear and Dagan (1963, 1964b) experimentally using a Hele-Shaw model, and also modeled numerically by Shamir and Dagan (1971) using the mesh regeneration finite differ-

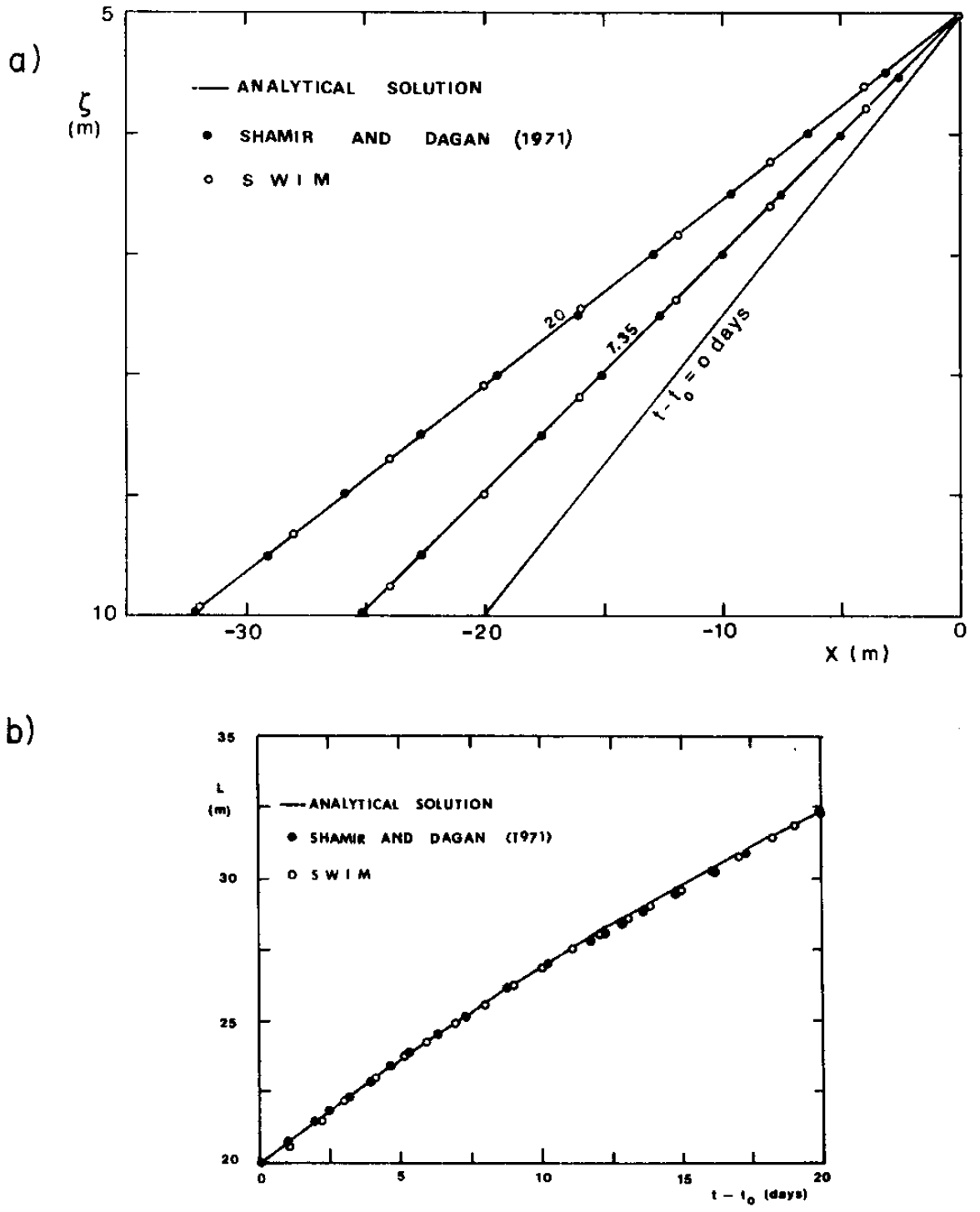


Figure 5.1 - Gravitational Segregation Problem: Comparison to the Mesh Regeneration Solution of Shamir and Dagan (1971).
a) Interface Movement; b) Toe Movement.

ence numerical code mentioned in Section 5.2. The experimental data reported in these publications was employed in the SWIM simulation.

Figure 5.2 presents a typical cross section for this problem with the actual dimensions of the Hele-Shaw experiments. The other relevant parameters reported are:

$b = 27$ cm

$\gamma^f = 1.0$ g/cm^3

$\gamma^s = 1.029$ g/cm^3

$K^f = 69$ cm/sec

$n = 1.0$

The SWIM simulation employed a first type specified head boundary condition for the seawater at the "coast" ($\phi^s_{x=0} = 0$), a second type specified flux boundary condition for the freshwater at the inland extreme ($Q^f_{x=400} = Q^f_\infty$ = specified), and a third type coastal boundary condition for the freshwater at the coastline ($K'_c = 69.0$ cm/sec). Two types of grids were used (see Figure 5.3); one using linear elements, and the other with the same number of nodes but using quadratic elements. The results obtained with both grids were identical and the comments made here are valid for both.

5.3.1 Advancing Seawater Wedge

In this simulation an initial ($t < 0$) flow per unit width of $Q^f = 19.1$ cm^2/sec was considered. Later ($t > 0$) this flow was suddenly cut off, and the advancement of the interface observed. This is Experiment 1 of Bear and Dagan (1963, 1964b). Two types of SWIM simulations

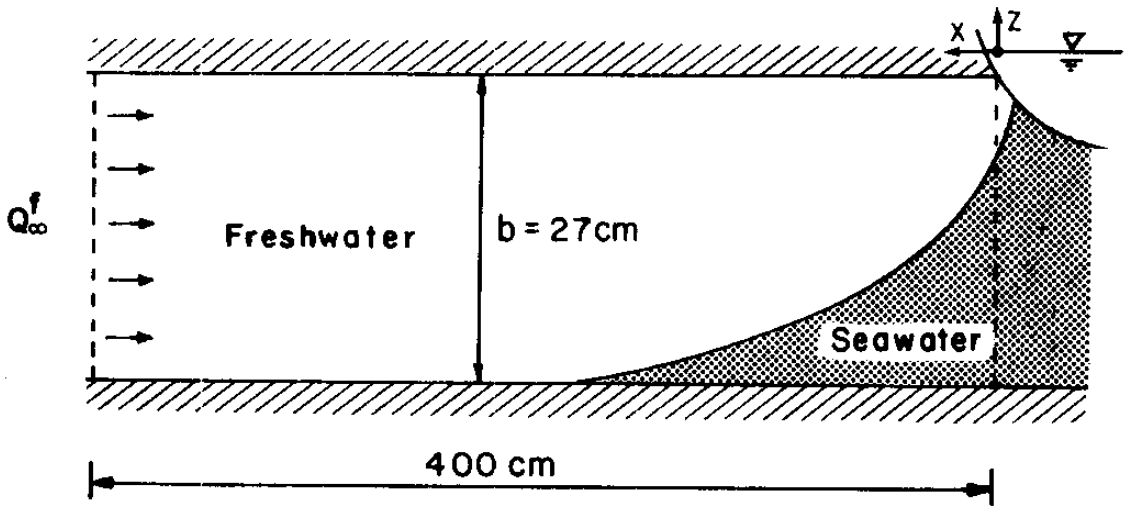


Figure 5.2 - Schematic Representation of the Classical 1-D Seawater Intrusion Problem

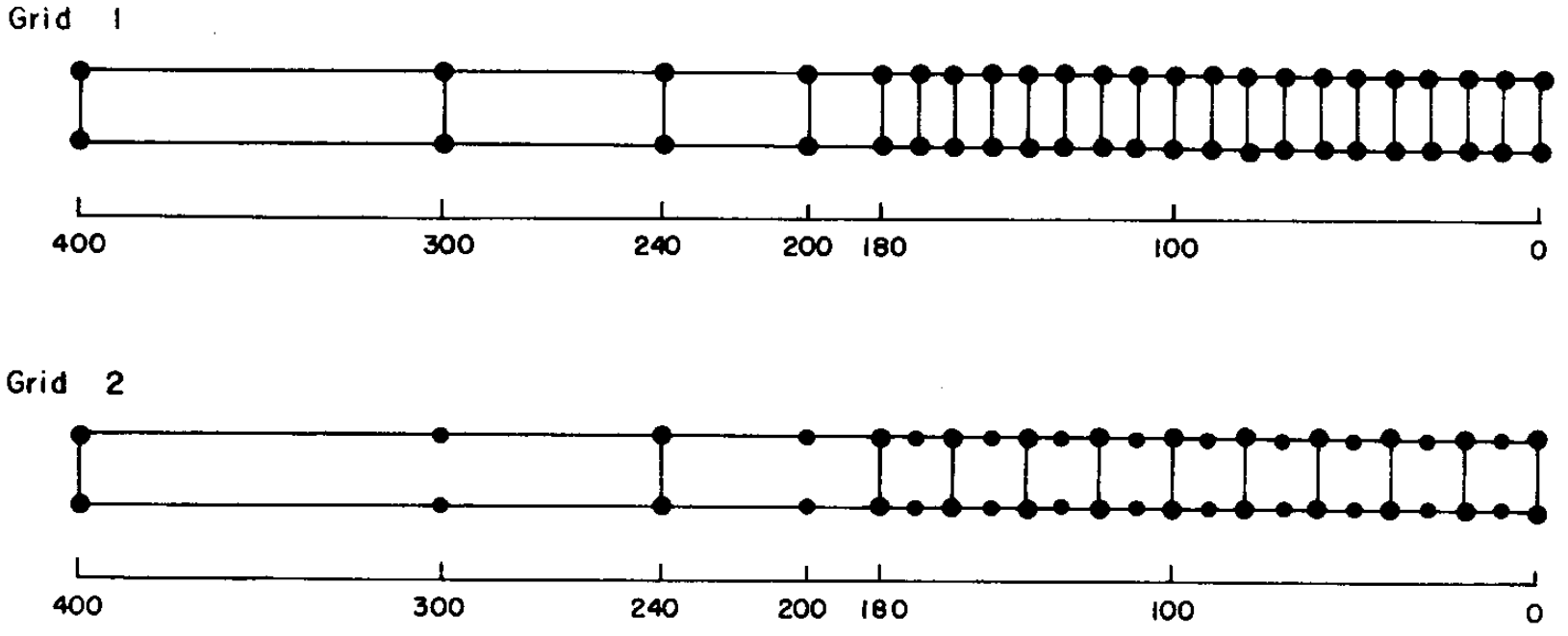


Figure 5.3 - Finite Element Grids Used in the Classical 1-D Seawater Intrusion Problem of Bear and Dagan (1963, 1964a)

were run for this experiment: one for the initial steady state condition, and the other for the transient solution. In both runs the coefficient for third type boundary condition at the sea was considered equal to the freshwater permeability, that is $K'_c = K^f = 69.0$ cm/sec.

For the steady state solution the results of two numerical runs and the Hele-Shaw data are presented in Figure 5.4 (see the lower right hand corner). The first numerical result is for $Q^f_\infty = 19.1$ cm^2/sec (empty squares on the figure), as reported by Bear and Dagan (1963, 1964b). The SWIM results roughly agree with the Hele-Shaw data, especially realizing that the Dupuit approximation is not completely valid for such a slope of the interface. The results of a second simulation with $Q^f_\infty = 24.0$ cm^2/sec (stars on the figure) are also presented; the match is even better; in fact, it is almost exact.

That a higher flow rate than the one reported by Bear and Dagan gives a better agreement with the data can be attributed to one of two sources: model (SWIM) error or experimental error. There is reason to believe that there may be some experimental error (see the next section), but little can be done to check it, and it is unlikely that it can account for the entire difference between $Q^f_\infty = 24.0$ and 19.1 cm^2/sec. Significant SWIM model error can stem from the Dupuit approximation or mis-identification of the third type boundary condition parameter at the coastline. Sensitivity analysis of this parameter was not attempted for this experiment; however, it was done for the retreating seawater wedge described in the following section.

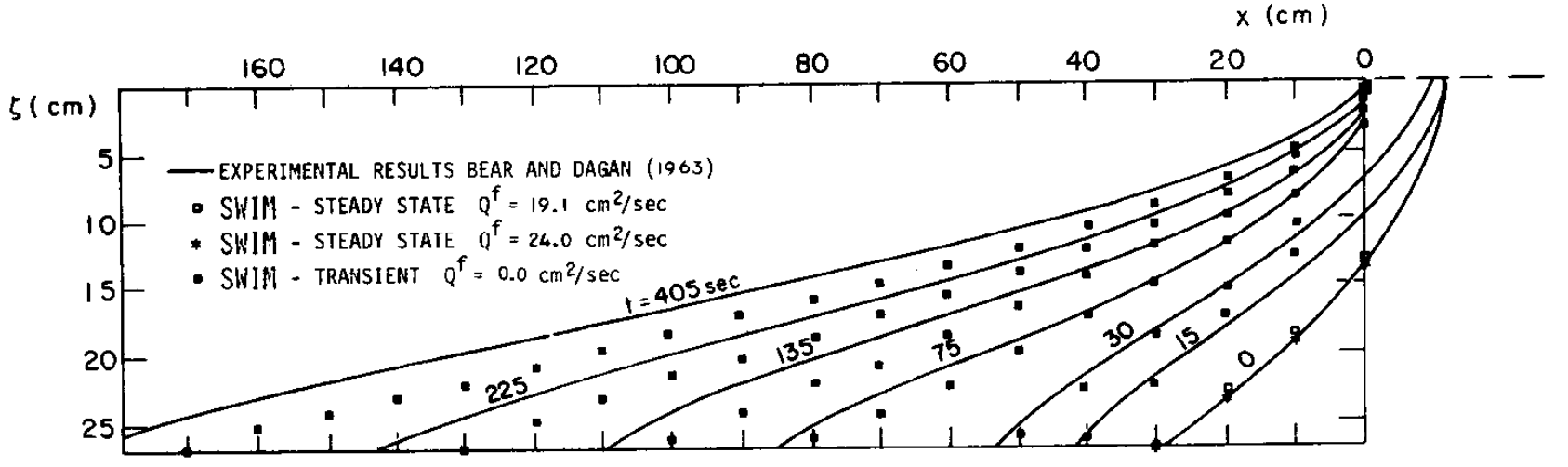


Figure 5.4 - Comparison Between Hele-Shaw Model Experimental Results (Bear and Dagan 1963, 1964a) and SWIM Solution for an Advancing Seawater Wedge

The transient run results are also presented in Figure 5.4. The general pattern of the SWIM simulation follows the pattern of the experimental data; however, in the numerical result the toe advances at a slower rate and the opening to the sea closes at a faster rate. These two phenomena are interdependent. The opening to the sea closes faster because the coefficient K'_c of the sea boundary condition is too high and the head loss at the coast is small. Since this opening to the sea is small when the seawater advances, the amount of freshwater that can flow to the sea is constrained, slowing down seawater intrusion.

A comparison to Shamir and Dagan's (1971) numerical results was not attempted because they present the interface solution using an artificial initial condition: the interface location at t = 45 sec. When they used the actual initial conditions, they observed a lag in the toe dispalcement between their model solution and the experimental data for t > 80 sec., probably similar to the lag observed in Figure 5.4. However, they have not published these results.

5.3.2 Retreating Seawater Wedge

According to Bear and Dagan (1963, 1964b), in their Experiment 3, the freshwater flow for $t \leq 0$ was $Q_\infty^f = 3.9$ cm^2/sec, which was increased to $Q_\infty^f = 18.8$ cm^2/sec for $t > 0$. For this data two SWIM simulations were performed: one to obtain the initial steady state solution, the other for the transient solutions. The steady state solution is presented in Figure 5.5 (the empty squares), and shows an excellent agreement with

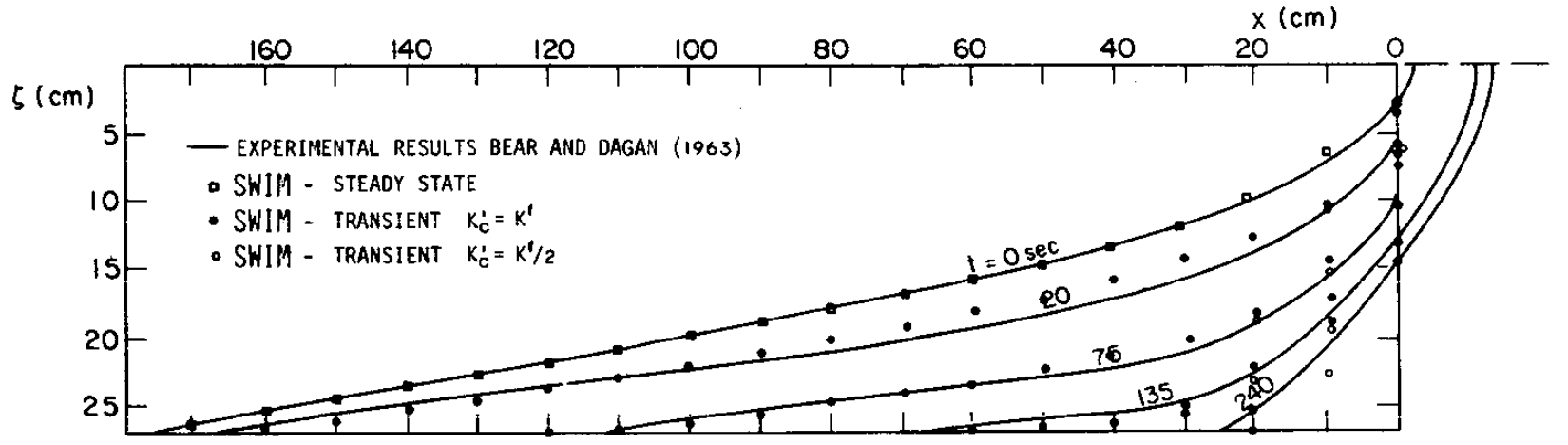


Figure 5.5 - Comparison Between Hele-Shaw Model Experimental Results (Bear and Dagan 1963, 1964a) and SWIM Solution for a Retreating Seawater Wedge ($Q_\infty^f\Big|_{t=0} = 3.9$ cm^2/sec; $Q_\infty^f\Big|_{t>0} = 30.0$ cm^2/sec)

the experimental data, with the exception of the zone with higher curvature where the Dupuit assumption is not particularly valid.

The transient solution is presented in the next figure, Figure 5.6, along with Shamir and Dagan's (1971) mesh regeneration solution Both numerical solutions show a strong agreement with each other, with the exception of the zone near the coast. This is due to the fact that in SWIM a third type coastal boundary condition is used, while in Shamir and Dagan the interface depth at x = 0 was imposed using the experimental data, data that was not available in the published literature. The most striking feature of Figure 5.6 is the complete disagreement between the numerical results (both models) and the experimental data. This difference deserves a closer look.

The volume (volume per unit width = area) between the Hele-Shaw interface at t = 0 and at t = 20 sec is approxiatemly 480 cm^2. Assuming that for $0 \leq t \leq 20$ sec the freshwater flow to the sea is constant and equal to the initial value $Q_\infty^f = 3.9$ cm^2/sec (an optimistic assumption, because this flow should increase with t), the net increase of the freshwater flow, $\Delta Q_\infty^f = Q_\infty^f\big|_{t=20 \text{ sec.}} - Q_\infty^f\big|_{t_o} = 18.1 - 3.9 = 14.2$ cm^2/sec, should account for the area swept by the interface in the first 20 sec. However,

$$\Delta Q_\infty^f \times 20 \text{ sec} = \text{max.vol. added} = 14.2 \text{ cm}^2/\text{sec} \times 20 \text{ sec} =$$
$$= 284 \text{ cm}^2 < 480 \text{ cm}^2 = \text{vol. displaced}$$

which means that the flow measured by Bear and Dagan of 18.1 cm^2/sec could not possibly produce the interface displacement observed. Re-

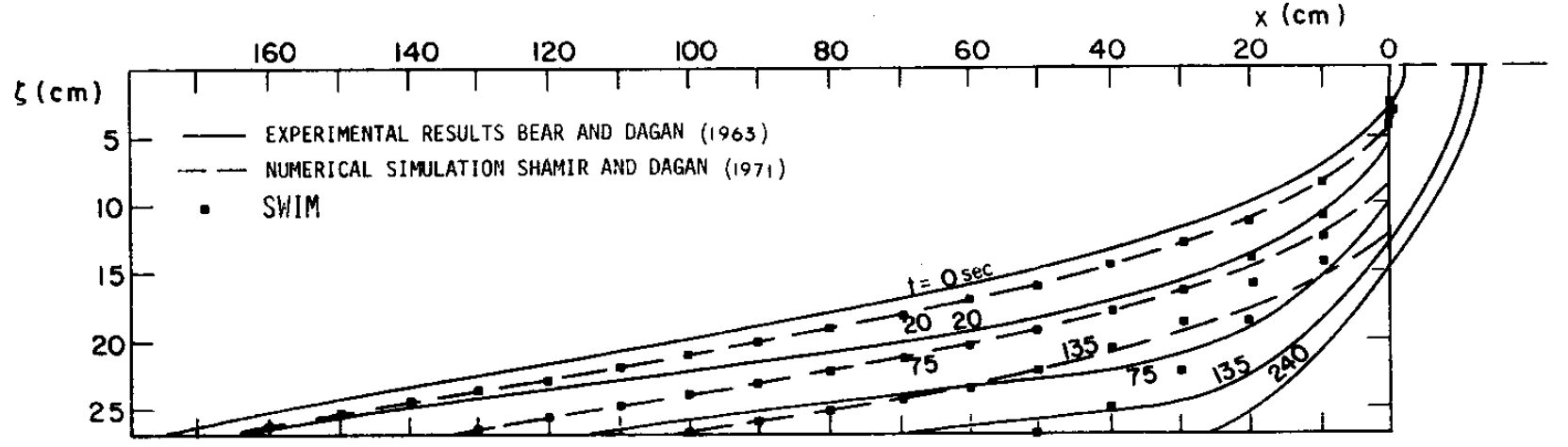


Figure 5.6 - Comparison of Hele-Shaw Model Experimental Results (Bear and Dagan 1963, 1964a), to Mesh Regeneration (Shamir and Dagan 1971) and SWIM Numerical Solutions for a Retreating Seawater Wedge ($Q_\infty^f\Big|_{t=0}$=3.9 cm^2/sec ; $Q_\infty^f\Big|_{t>0}$=18.1 cm^2/sec)

arranging the calculations leads to an estimate of the average freshwater influx during the 20 second period:

$$\hat{Q}^f_\infty \geq \frac{480}{20} + 3.9 = 27.9 \text{ cm}^2/\text{sec}$$

Since the flow to the sea was assumed constant, and equal to the initial value, and since in reality it increases, it was decided to assume a value of $\hat{Q}^f_\infty = 30 \text{ cm}^2/\text{sec}$ as appropriate for this experiment, and to compare SWIM's results for this case with the experimental data. This comparison is shown in Figure 5.5, the black circles represent the SWIM simulation. Now the agreement between the numerical and experimental results is much better and again the differences are near the coastal opening to the sea. To correct this last problem another simulation was performed using a new parameter for the third type coastal boundary condition, $K'_c = K^f/2$, that is half of the value used in the previous runs. The results are slightly improved at the nodes near the coast, as expected, and are shown in Figure 5.5 as the empty circles. The overall agreement between this last simulation and the experimental data is excellent, especially for $t \leq 75$ sec. Some differences appear for later times that indicate that the freshwater inflow in the experiment was actually time dependent. All of these findings confirm the initial hypothesis, of errors in the collection of experimental data.

Dagan (personal communication, 1979) has indicated that the measurement of the flows was relatively crude, and that the different flow rates were obtained by opening and closing values to constant head

tanks positioned at different levels. This alone would indicate experimental difficulty in controlling a constant freshwater inflow. He also indicated that it was very difficult to maintain a constant interspace width between the walls of the Hele-Shaw model, leading to errors in calculating the model permeability, specific discharge, and discharge per unit width, Q_x^f or Q_x^s.

Through these simulations SWIM has proved that it can handle both advancing and retreating wedges. The third type coastal boundary condition has worked remarkably well, although it requires application specific calibration. For the drastic changes in the opening to the sea observed in these experiments, a scheme with a nonlinear third type boundary condition at the coast might be preferable. Such a nonlinearity would include a freshwater head dependent coefficient,

$$K_c' = K_c' (\phi^f)$$

5.4 Development of a Lens of Freshwater Over Seawater

SWIM was designed to handle situations in which the interface between freshwater and seawater intersects the bottom of the aquifer in which a "toe" exists. However, it can also be used to simulate situations in which freshwater floats as a lens over the seawater, so that there is no "toe". To judge the model performance in this situation, as well as to infer the influence of the position of the aquifer bottom on the shape of the interface, the transient development of a freshwater lens was simulated. Approximate analytical solutions have

been proposed by Hantush (1968) for thick aquifers near the coast or under islands. From these solutions, the example of a coastal aquifer, with finite width L, receiving uniform recharge from an infinitely long strip parallel to the coast, was selected (see Figure 5.7). The depth of the interface according to Hantush (1968) is:

$$\begin{aligned}
\zeta^2 &= 2c[2(b-a)x-f(x,t)] \quad \text{for } 0 < x < a \\
&= 2c[2(b-a)x-(x-a)^2-f(x,t)] \quad \text{for } a < x < b \\
&= 2c[(b^2-a^2)-f(x,t)] \quad \text{for } b < x < L
\end{aligned} \tag{5.1}$$

in which

$$f(x,t) = \left| \left(\frac{32L^2}{\pi^3}\right) \sum_{n=0}^{\infty} R_n \sin \frac{(2n+1)\pi x}{2L} \right|$$

$$R_n = \frac{1}{(2n+1)^3} \left[\cos \frac{(2n+1)\pi b}{2L} - \cos \frac{(2n+1)\pi a}{2L}\right] \exp\left\{-\left[\frac{(2n+1)\pi}{2}\right]^2 \frac{\bar{\nu} t}{L^2}\right\}$$

$$c = \frac{N}{2\delta(1+\delta)K^f}$$

$$\bar{\nu} = \frac{K^f \delta \bar{\zeta}}{n}$$

$$\delta = \frac{\gamma^s - \gamma^f}{\gamma^f}$$

in which:

$|(\ )|$ = absolute value*;

---

*The absolute value of f(x,t) is not included in Hantush (1968), but for certain cases singularities appear if the absolute value is not used.

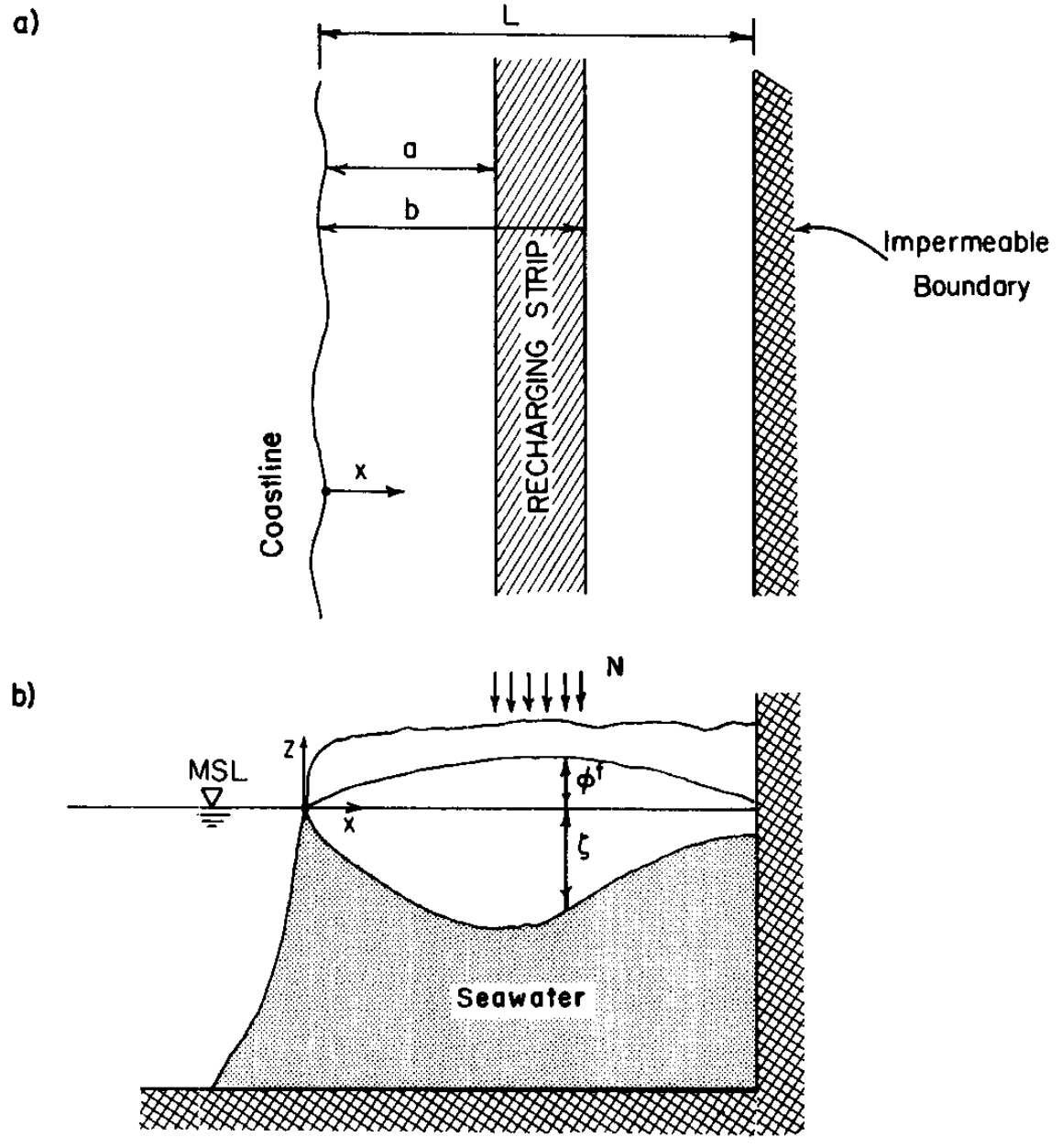


Figure 5.7 - Schematic Representation of the Development of a Freshwater Lens Over Seawater. a) Plan View; b) Cross Section.

N = recharge, or accretion, rate per unit of area;

K^f = permeability;

n = effective porosity;

$\bar{\zeta}$ = average depth of the interface = $\bar{\zeta}(t)$.

a,b, L and x are defined in Figure 5.7. The discharge per unit width of the sea coast in given by:

$$Q^f_{x=0} = -N\left[(b-a) + \left(\frac{8L}{\pi^2}\right)\sum_{n=0}^{\infty}(2n+1)\,R_n\right] \qquad (5.2)$$

In his derivation Hantush assumed that the Dupuit approximation is valid, used the Gyben-Herzberg hypothesis, thus neglecting flow in the seawater phase, and employed a second type linearization of the resulting equation (see Bear, 1972, for more on linearizations of the Dupuit type equations).

For the SWIM simulations of this situation the following parameters were selected:

L = 100. m

a = 45. m

b = 55. m

N = 0.2 m/day

$K=K^f$ = 100. m/day

$K'_c = K^f$

γ^f = 1.0

γ^s = 1.025

The finite element grid used in this problem employs linear 4-node

elements with a length of 5 m, similar to grid 1 of Figure 4.18. At the left-hand boundary a third type freshwater coastal boundary condition was imposed and first type seawater boundary condition $\phi^s\big|_{x=0} = 0$. A "no flow" condition was imposed on the right hand boundary. An arbitrary depth of the aquifer of 20 m was selected for the simulations; the effect of changing this depth will be examined later. The initial condition for the interface and phreatic surface was specified as a horizontal plane with elevation equal to the mean sea level.

In his development Hantush assumed a static seawater level ($\phi^s = 0$, all x). This was simulated in SWIM by selecting a extremely large hydraulic conductivity for the seawater, $K^s/K^f = 10^4$.

Figure 5.8a presents the results from the SWIM simulation, solid line, and the analytical solution (Eq. 5.1), dashed line. At early times the Hantush solution does not conserve mass, as observed by comparing the sum of the volume of freshwater lens plus the outflow to the sea, to the total recahrged volume, V = N(b-a)t. Later, for $t \leq 100$ days, the Hantush solution is conservative. The explanation for this minor mass conservation problem lies in the way the averaged interface depth ($\bar{\zeta}$) is calculated and used in Eq. 5.1. The SWIM solution is conservative at all times. The final steady state solution for both cases is identical, as one should expect, because the Ghyben-Herzberg assumption is applicable to both cases, and the linearization for the analytical solution is no longer necessary. The principal explanation for the observed difference between solutions during the transient can be attributed to these added approximations of the

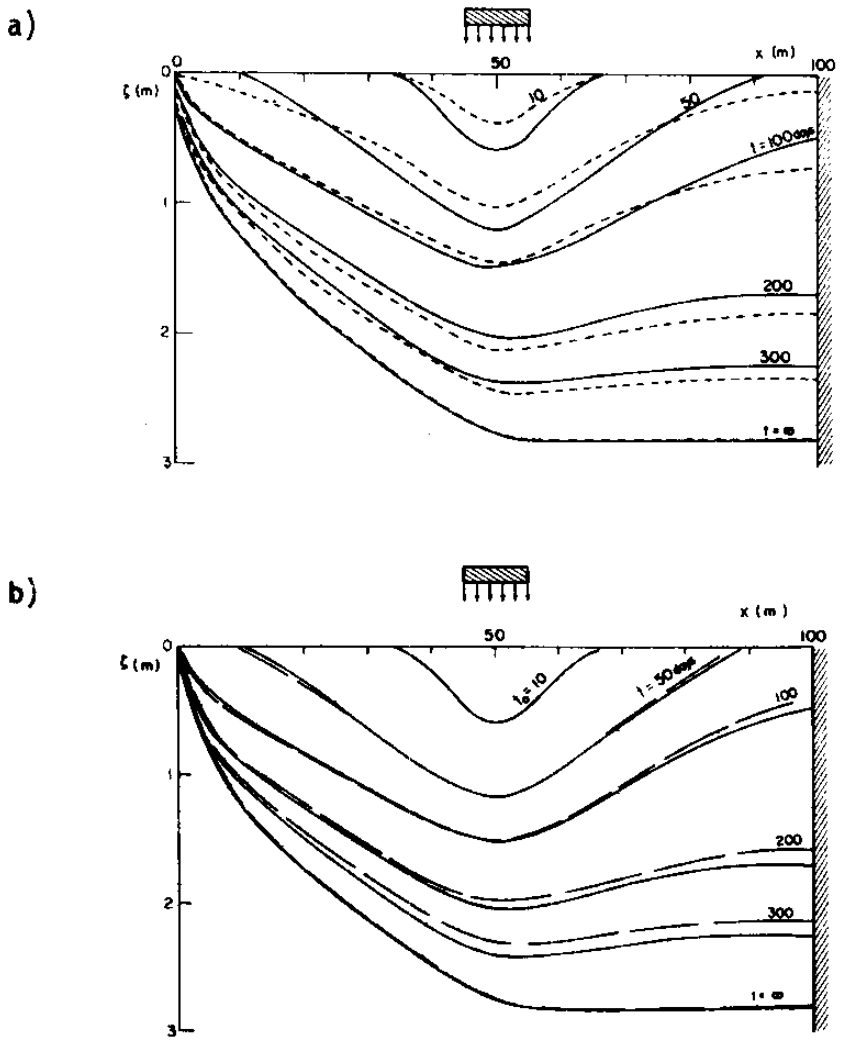


Figure 5.8 - Interface Movement in the Development of a Freshwater Lens over Seawater: a) Comparison Between Hantush (1968) and SWIM; b) Effect of the Ghyben-Herzberg Assumption.

analytical solution: Ghyben-Herzberg and the linearization. Figure 5.9a shows the drawdown beneath the midpoint of the recharge area; as one can see both solutions are similar. The outflow per unit of width to the sea computed by SWIM and by using Eq. 5.2 is compared in Figure 5.10a. The outflow computed by Hantush's method is slightly smaller; consistent with its faster lens development.

To infer the influence of the Ghyben-Herzberg assumption a simulation assuming $K^s \approx K^f$ was performed. The results are compared with the previous simulation, in which $K^s/K^f = 10^4$, of Figure 5.8b. The final steady state solutions are identical. The transient interface that accounts for seawater dynamics moves slower than when the seawater is assumed static (see Figure 5.9b), although the freshwater discharge to the sea does not change much (see Figure 5.10b). In fact, during the first 40 to 50 days of lens development there is almost no difference in the solution. The lens is less than one meter thick, and when this is compared to the overall aquifer thickness of 20 m, the Ghyben-Herzberg approximation is reasonable. Later, however, the lens thickness grows and the approximation becomes less valid. Also note, for this example that the additional error of the linearization is as large as, or larger than, the error due to the Ghyben-Herzberg approximation.

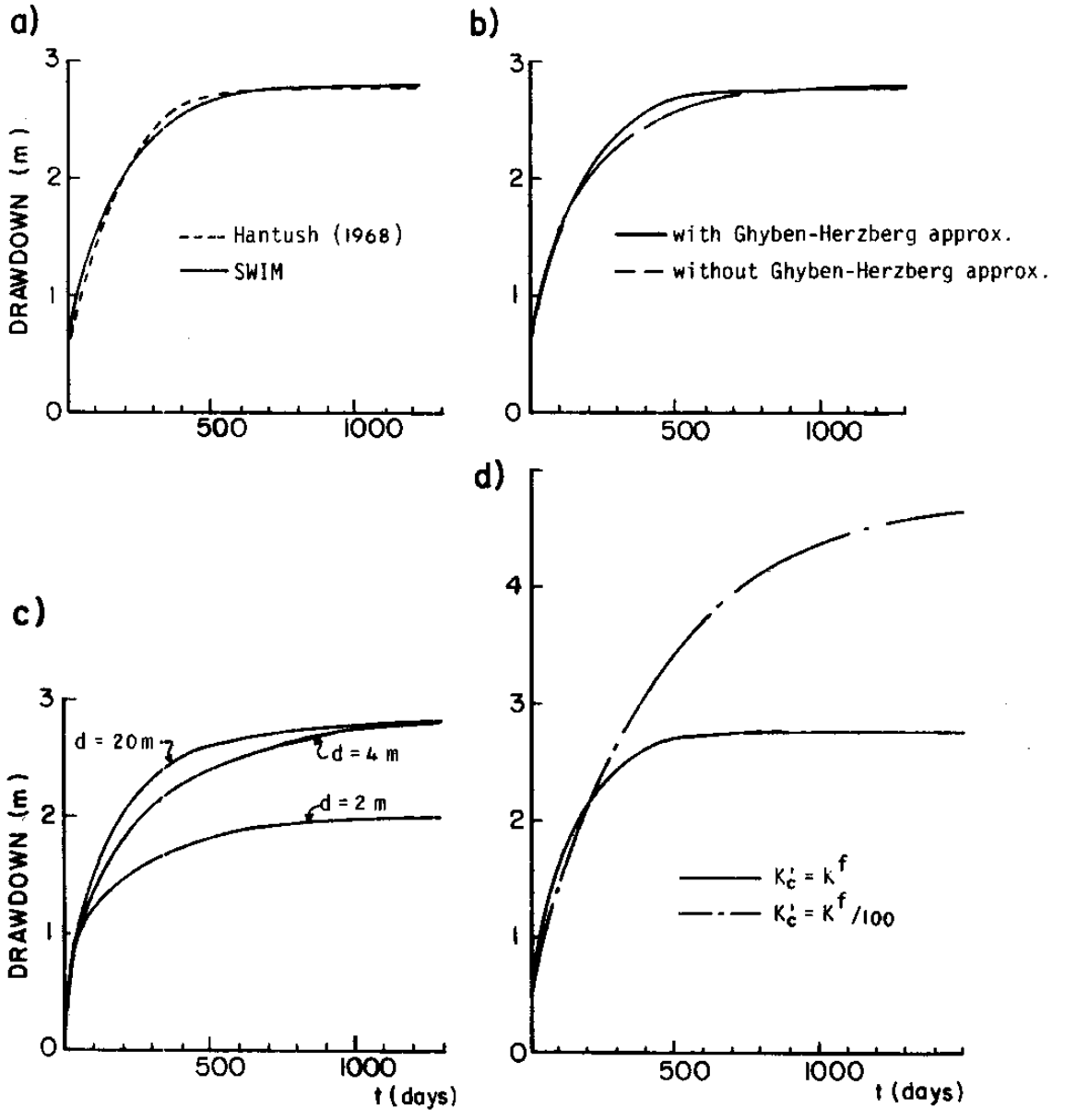


Figure 5.9 - Comparison of the Drawdown at the Middle Point of the Freshwater Lens for Different Situations: a) Hantush (1968) Versus SWIM;
b) Effect of the Ghyben-Herzberg Assumption;
c) Influence of the Bottom Elevation;
d) Influence of the Coast Boundary condition Coefficient

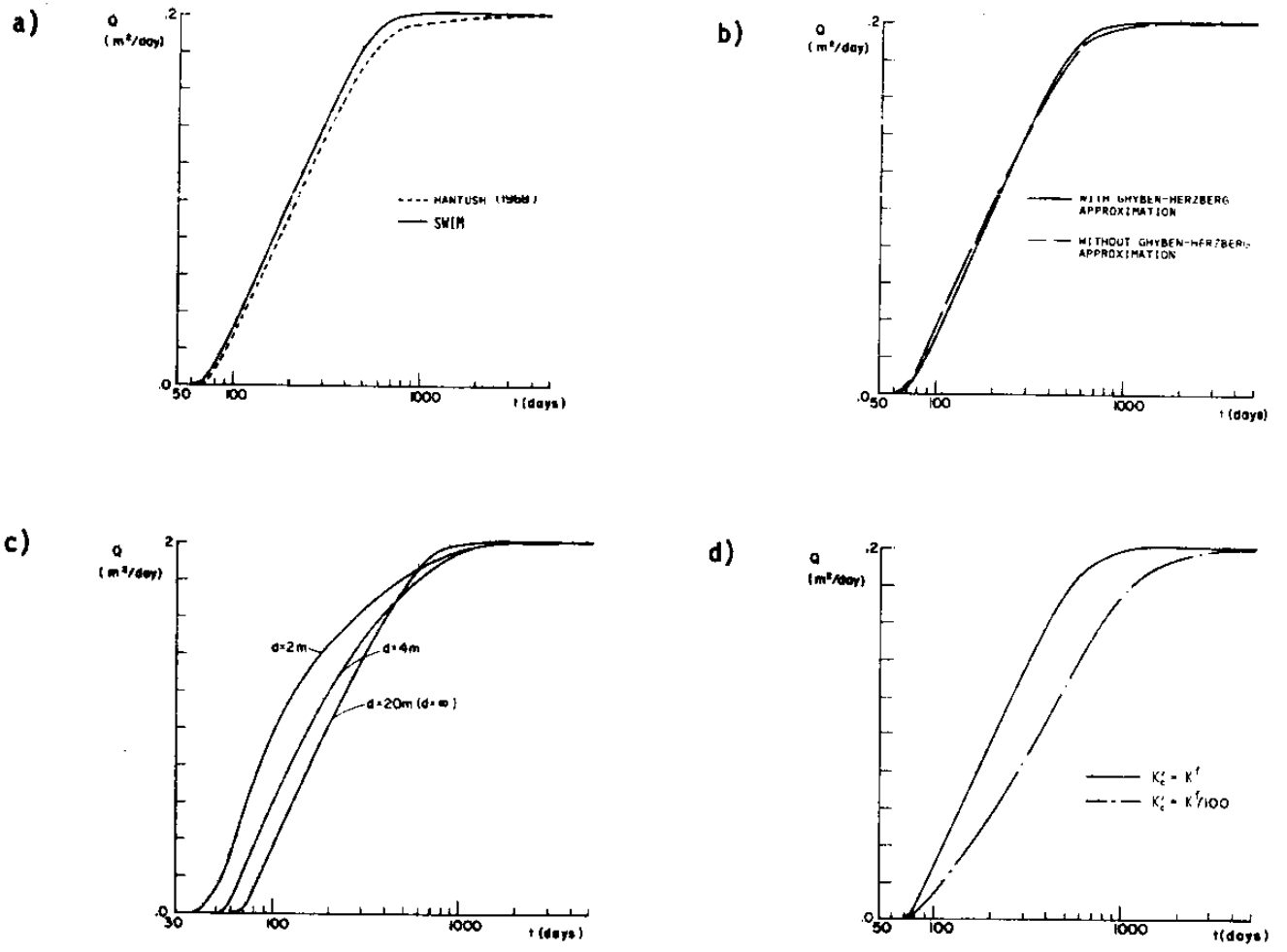


Figure 5.10 - Comparison of the Freshwater Flow to the Sea for Different Situations of the Development of a Freshwater lens over Seawater: a) Hantush(1968) vs. SWIM; b) Effect of the Ghyben-Herzberg Assumption; c) Influence of the Bottom Elevation; d) Influence of the Coastal Boundary Condition Coefficient.

The appropriateness of the Ghyben-Herzberg approximation is related to the aquifer depth. In the solutions presented thus far the maximum depth of the interface has been less than 3 m, which is much smaller than the total depth of the aquifer, 20 m. In these examples, if one can consider the aquifer to be a thick formation, then the Ghyben-Herzberg should apply. In the next set of simulations the seawater was assumed dynamic ($K^s \simeq K^f$), while the elevation of the bottom of the aquifer was varied from depths of 20 m to 4 m, and to 2 meters. The interface positions for these three simulations are presented in Figure 5.11, the drawdown at the middle part of the recharge zone is shown in Figure 5.9c and the unit width freshwater discharge to the sea is given in Figure 5.10c.

The influence of the bottom is felt in two important ways: a reduction of the rate of drawdown of the interface, and an increase of the freshwater discharge to the sea. These two phenomena are related and are due to the resistance the seawater encounters when it attempts to flow back from the aquifer to the sea. An interesting note is that for the 2 m depth no seawater is trapped in the aquifer by the freshwater lens, when it hits the bottom. As expected the steady state shape of the interface is the same for all the three cases.

This example clearly demonstrates that in many cases the Ghyben-Herzberg assumption of a infinitely deep aquifer, to avoid the problem of modeling the sea water regime, is not always satisfactory, unless one is seeking a steady state solution. Also, the Ghyben-Herzberg approximation is equivalent to either, increasing K^s/K^f to a large

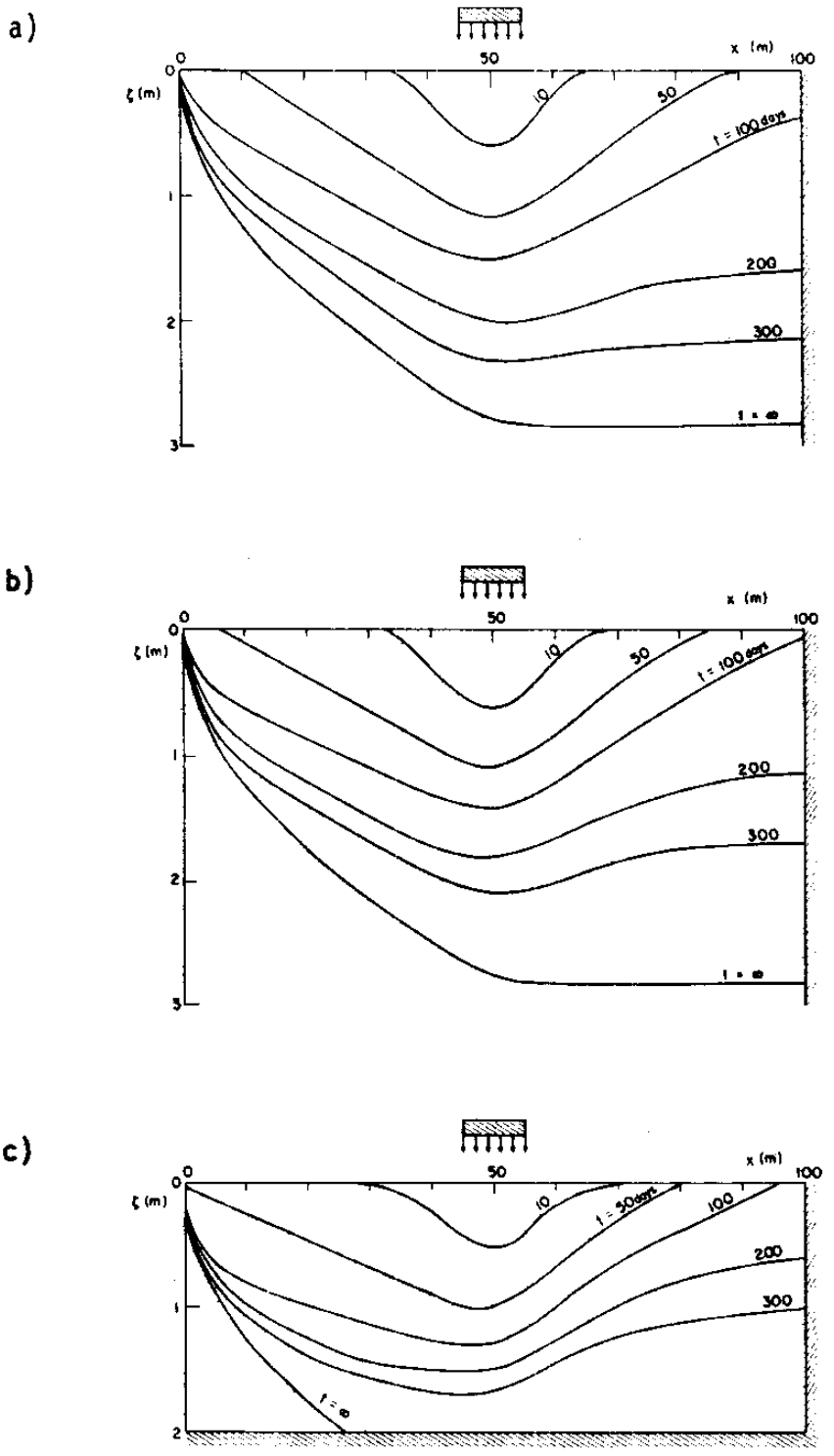


Figure 5.11 - Interface Movement in the Development of a Freshwater Lens over Seawater. Influence of the Bottom Elevation.
a) Depth of the Bottom d = 20 m ; b) d = 4 m ; c) d = 2 m .

number, or increasing the aquifer thickness; both result in essentially zero resistance to flow in the seawater regime.

As a final test using this example, a sensitivity analysis was performed for the outflow coefficient of the freshwater coastal boundary condition. So far, a coefficient of $K_c' = K^f$ has been used; new simulations assuming $K_c' = K^f/10$ and $K'^f = K^f/100$ were performed. For $K_c' = K^f/10$ the results were identical to the previous simulation with only a slight change at the coastal freshwater opening. For $K_c' = K^f/100$ there was a significant effect for $t > 100$ days, especially near the coast, and for the final steady state solution, as shown in Figure 5.12. The creation of the almost uniformly thick lens apparent in the figure is due to the great resistance the freshwater flow across the coastal "third-type barrier". In Figure 5.9d, the drawdown at the middle point of the recharge is plotted. It shows a divergence from the basic solution only for $t > 200$ days, after the interface reaches the coast. In Figure 5.10d the unit width freshwater discharge to the sea shows a deceleration of the discharge, because the freshwater is being used to create this enormous lens. Clearly, this very small value of $K_c' = K^f/100$ is unrealistic, and does not account for vertical flow/nonDupuit effects observed near the coast. $K_c' \simeq K^f$ seems to perform adequately.

These numerical experiments demonstrate the capability of SWIM to model freshwater lens situations. They further confirm the use of the

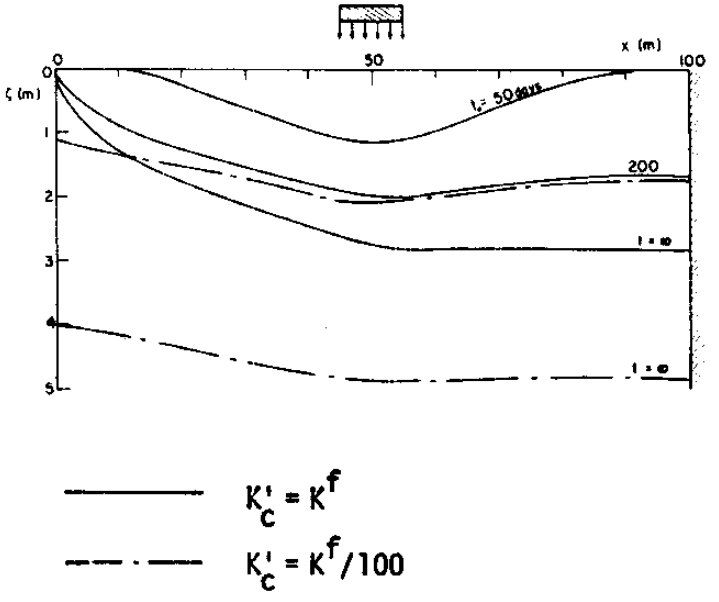


Figure 5.12 - Interface Movement in the Development of a Freshwater Lens over Seawater. Influence of the Coastal Boundary Condition Coefficient

third-type freshwater coastal boundary condition, and indicate that the solution is not too sensitive to its parameter, for a reasonable range of parameter values.

Comparison of the simulation results to Hantush's (1968) approximate solution, clearly points out the danger of using the Ghyben-Herzberg approximation and equation linearizations for this dynamic problem.

5.5 Seawater Intrusion in a Leaky Coastal Aquifer

In the previous simulations the leaky terms were used to represent the coastal boundary. In this test leakage terms of the governing equation are used to represent a leaky layer at the top of the aquifer. Consider the cross-sectional view of the leaky coastal aquifer shown in Figure 5.13. Freshwater is recharged at the left hand side, and from the upper aquifer ($\phi^f < \phi'^f$). Later, because of the decreasing value of ϕ'^f near the coast, the freshwater leaks back to the upper aquifer ($\phi^f > \phi'^f$). This problem has been solved analytically for the 1-D steady state case, by Hashish et al. (1979) using numerical integration on the following partial differential equation (Verou, 1978):

$$\frac{d^2\left(b^f\right)^2}{dx^2} - 2\,\frac{b^f K'}{b' K^f} = \frac{2K'}{b' K^f}\left(d - \frac{\gamma^f}{\Delta\gamma}\,\phi'^f\right) \qquad (5.3)$$

where:

b = aquifer thickness

b^f = freshwater thickness

b' = semi-pervious layer thickness

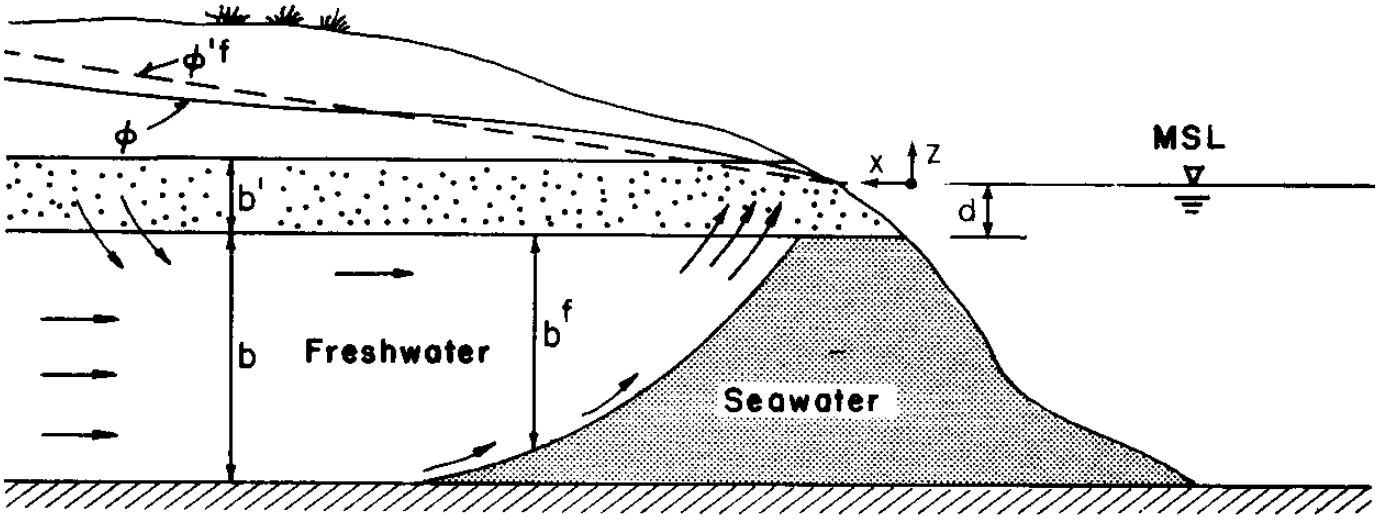


Figure 5.13 - Schematic Representation of Seawater Intrusion in a Leaky Coastal Aquifer

d = elevation of bottom of semi-pervious layer below MSL

K^f = aquifer permeability

K' = semi-pervious layer permeability

$\Delta\gamma = \gamma^s - \gamma^f$

$\phi'^f = \phi'^f(x)$ = freshwater piezometric head above the semi-pervious layer

The parameters selected for the SWIM simulation of this situation are identical to those used by Hashish et al. (1979) in a simple example they presented:

$b = 200$ m

$b' = 55$ m

$d = 40$ m

$K^f = 155$ m/day

$K'^f = 0.03$ m/day

$\gamma^f = 1.0$ g/cm^3

$\gamma^s = 1.025$ g/cm^3

$\phi'^f(x) = 10^{-4}\,x$, a function of space

$\ell = 120$ Km = length of the aquifer

For boundary conditions the saltwater head at $x = 0$ was set to zero ($\phi^s\big|_{x=0} = 0$) and the freshwater head at $x = \ell$ was assumed to be 10.0 m ($\phi^f\big|_{x=\ell} = 10.0$). A no flow condition was applied elsewhere, except that if the freshwater zone were to reach the coastline, a third-type boundary was specified. A linear grid with 10 km spacing was used in SWIM.

The results of the simulation and the numerically integrated analytical solution are presented in Figure 5.14. They cannot be distinguished at the scale of the graph. Table 5.1 gives the toe location for both solutions; again the results are almost equivalent.

Table 5.1

Toe Location Comparison for the Leaky Coastal Aquifer

	Toe Location (m)		Length of the Interface (m)
	Upper	Lower	
Hashish et al. (1979)	8.86	59.47	50.61
SWIM	9.16	59.18	50.02

With this simulation, SWIM's ability to model leaky aquifers was verified. It is curious, however, that the interface solution of the particular problem should turn out to be an almost straight line. Linearized solutions for this or similar problems have more "shape" to them (Verou, 1978; Hashish et al., 1979).

5.6 Injection of Freshwater into a Saline Aquifer

The injection of a fluid into an aquifer saturated with another fluid has become a problem of some interest in groundwater hydrology. This section analyzes the case of radial flow from a fully penetrating well in a horizontal confined aquifer with constant thickness. The injected fluid has a different density than the ambient fluid. Thus, this problem is similar in some ways to the one-dimensional gravitation-

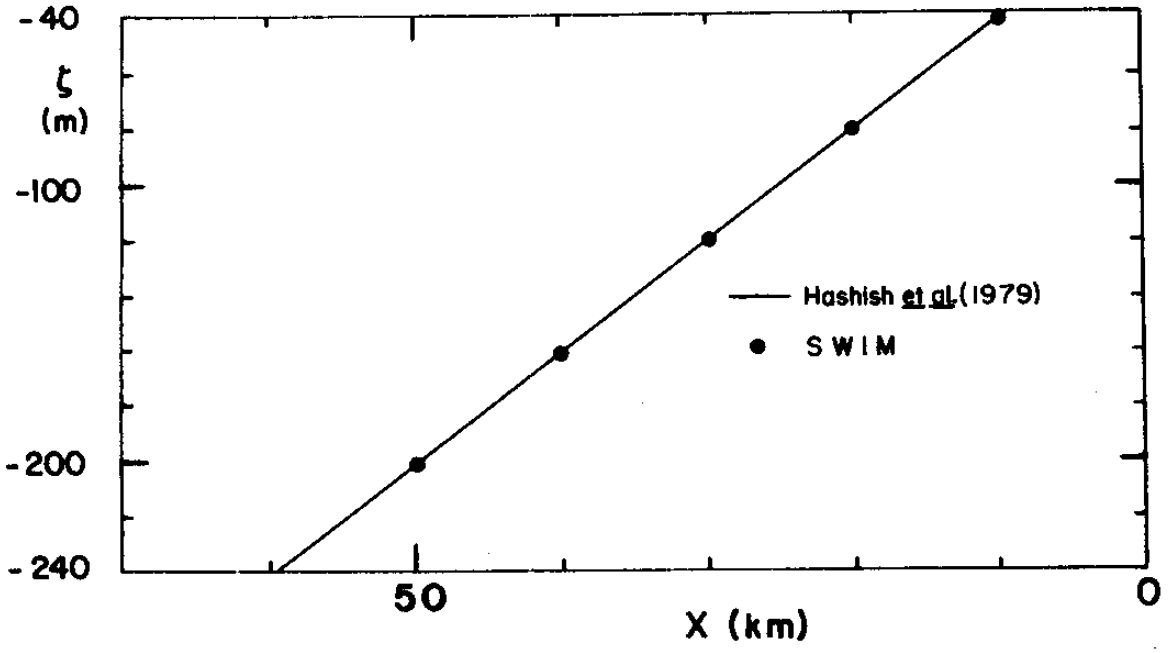


Figure 5.14 - Steady State Position of the Interface of the Seawater Intrusion in a Leaky Coastal Aquifer

al segregation problem of Chapter 4. However, in this case the interface does not remain linear in shape and, furthermore, it is radially displaced by the injected fluid. Gelhar, Wilson and Miller (1972) (see also Gelhar et al.,1972a) studied this problem and developed an analytical expression for the interface shape, which is used herein.

Figure 5.15 illustrates the process to be modeled. In a horizontal confined aquifer of constant thickness b the seawater, the heavier fluid, is displaced by freshwater, the lighter fluid. The analytical solution was derived in terms of radial distance (r) and volumetric coordinates (Ψ):

$$\frac{b^s}{b} = \frac{\eta}{b} = \frac{1}{2}(1 + U) + \frac{\varepsilon_r}{16}(1 - 3U^2) - \frac{\varepsilon_r^2 U}{112}(1 - \frac{51}{4}U^2) + \ldots \qquad (5.4)$$

where:

$\eta = b^s$ = interface elevation above the aquifer bottom [L]

b = aquifer thickness [L]

$U = \frac{\bar{\Psi}}{\varepsilon_r \Psi_*}$

$\bar{\Psi} = \Psi - \Psi_*$ $[L^3]$

$\Psi = \pi\, n\, b\, r^2$ $[L^3]$

$\Psi_* = Q\, t$ $[L^3]$

n = effective porosity

r = radial distance [L]

Q = recharge rate $[L^3/T]$

t = time [T]

$\varepsilon_r = \sqrt{D/A_r}$ = small perturbation parameter

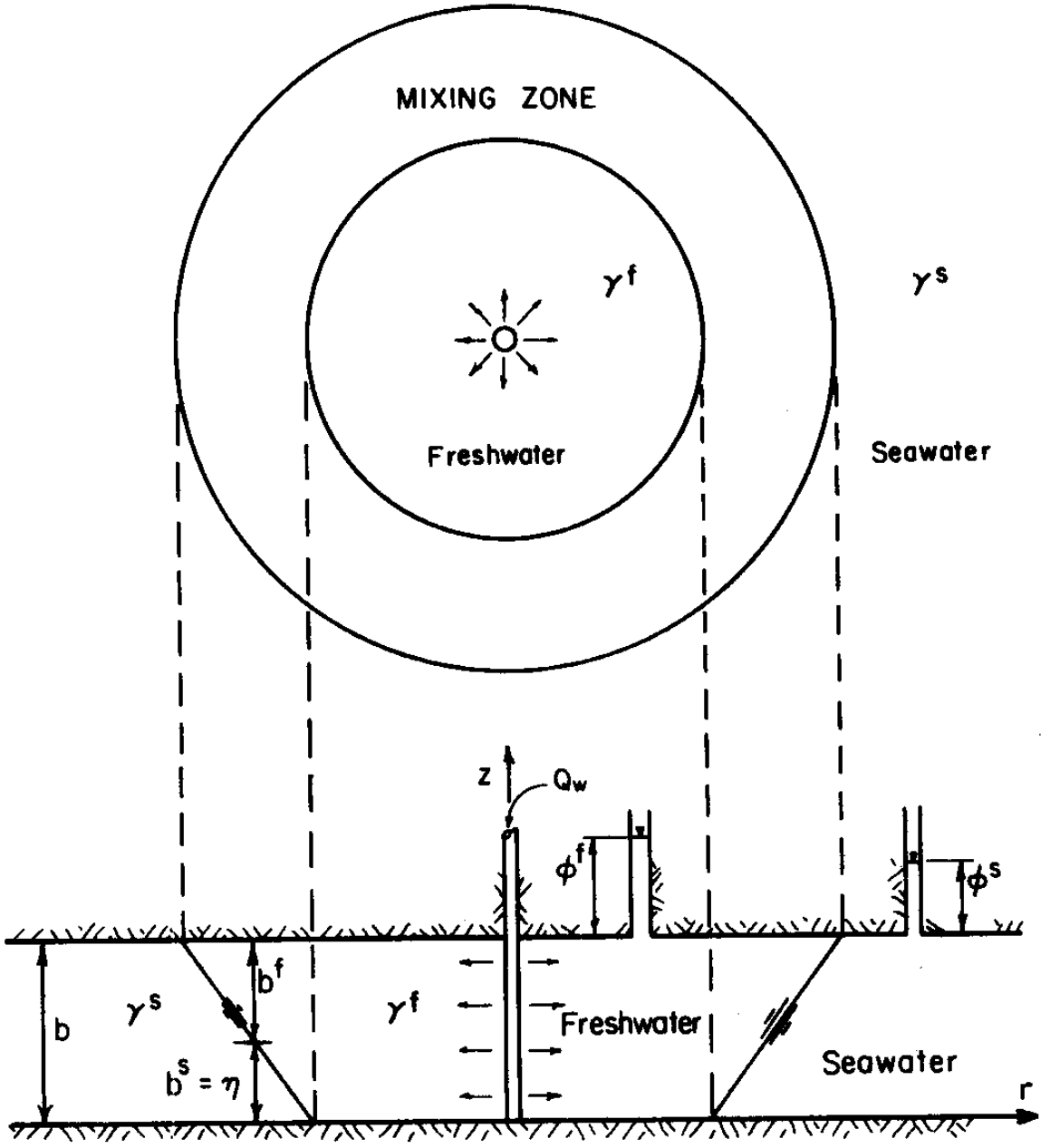


Figure 5.15 - Schematic Representation of Radial Freshwater Injection into a Saline Aquifer

$$A_r = \frac{Q}{2\pi nb} \quad [L^2/T]$$

$$D = \frac{K}{n}\frac{\Delta\gamma}{\gamma^f} b \quad [L^2/T]$$

$K = K^f$ = permeability [L/T]

This analytical solution assumes an initially vertical interface. However, this would require a very detailed grid, for the numerical simulation, and a corresponding very small time step to capture the earlier displacement of the interface. Since this earlier displacement is of little interest for present purposes, an initial condition at t_o = 10 days was calculated using Eq. 5.4.

For the numerical simulation the following parameters and boundary conditions were used:

b =	1.0 m	
K^f =	100 m/day	
n =	0.2	
γ^f =	1.0 g/cm^3	
γ^s =	1.025 g/cm^3	
Q_w =	1.859 m^3/day at x = 0.0 m	($x_1 = x_2 = 0.0$ m)*
Δt =	0.5 days	(Δt = 2.5 days)*
Δx =	1.0 m	($\Delta x_1 = \Delta x_2$ = 2.0 m)*
ϕ^s =	2.0 m at x = 24.0 m	(x_1 = 24 m, $0 \leq x_2 \leq 24$ m and x_2 = 24 m, $0 \leq x_1 \leq 24$ m)*
t_o =	10 days	

*The values inside parenthesis were used in the 2-D simulation described below.

Two simulations using linear elements were performed using SWIM. In the first simulation a 1-D radial flow grid shown in Figure 5.16 was employed; the results obtained are plotted in Figure 5.17. The agreement between the analytical and numerical solution is very good, although the numerical simulation shows a small acceleration of the upper toe. This may be due to the influence of the boundary conditions or because of the lack of higher order terms in the analytical solution.

Thus far SWIM has only been tested against essentially 1-D problems, whether linear or radial in geometry. This radial flow application can be looked upon as a two-dimensional problem, if a non-symmetric grid is employed, such as the one shown in Figure 5.18. This grid does not take advantage of the variable grid feature of the finite element method to provide more detail near the well. Thus it is also interesting to see how well the model performs with such an adverse discretization. The simulation results are given in Figure 5.19, with interface levels as computed along the two sides AB and AC, and along the diagonal AD, of the grid. The side based solutions show a small assymetry that may be due to the way the nodes were numbered combined with the column equation solver and the closeness of the boundaries. These solutions show the same acceleration on the upper toe noticed for the radial grid. The solution along the diagonal demonstrates a much better agreement with the theory, for the entire interface including the upper toe. The upper toe at t = 50 days along the diagonal of the grid is further from the sides of the grid at which the first type boundary conditions are applied. It shows no acceleration. This reinforces the earlier hypothe-

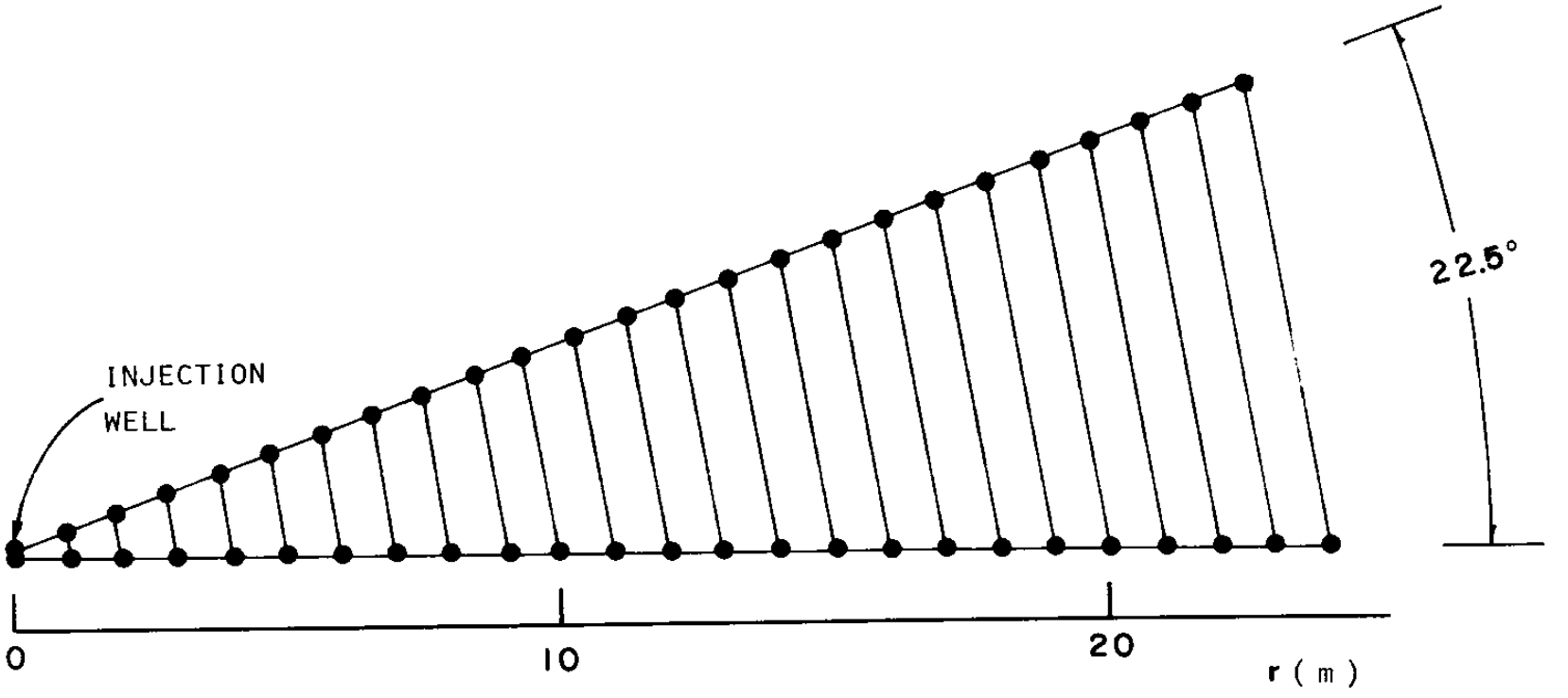


Figure 5.16 - Radial Finite Element Grid Used in the Injection of Freshwater into a Saline Aquifer Problem

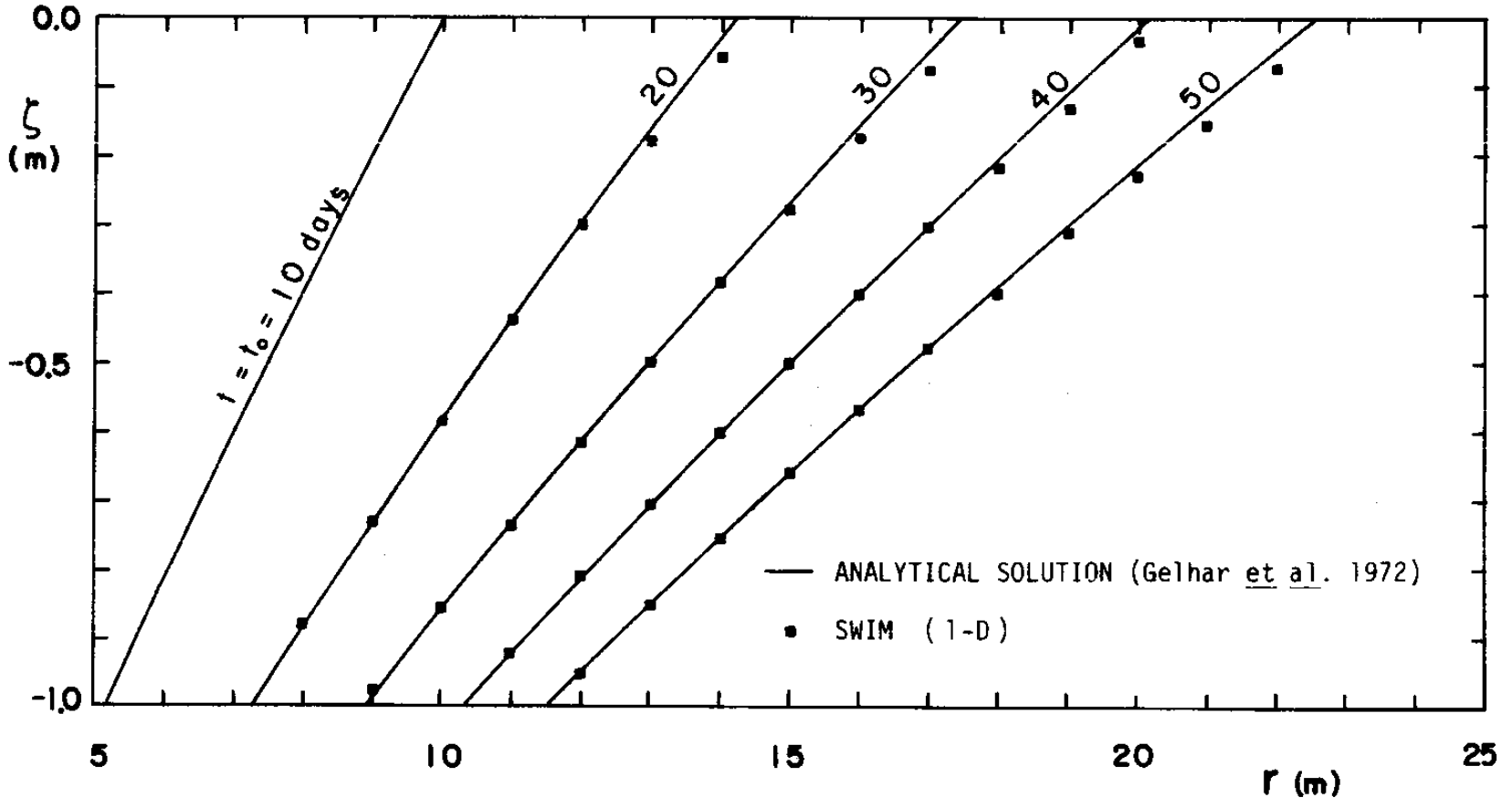


Figure 5.17 - Interface Movement in the Injection of Freshwater into a Saline Aquifer. Analytical Versus SWIM Radial Grid Solutions

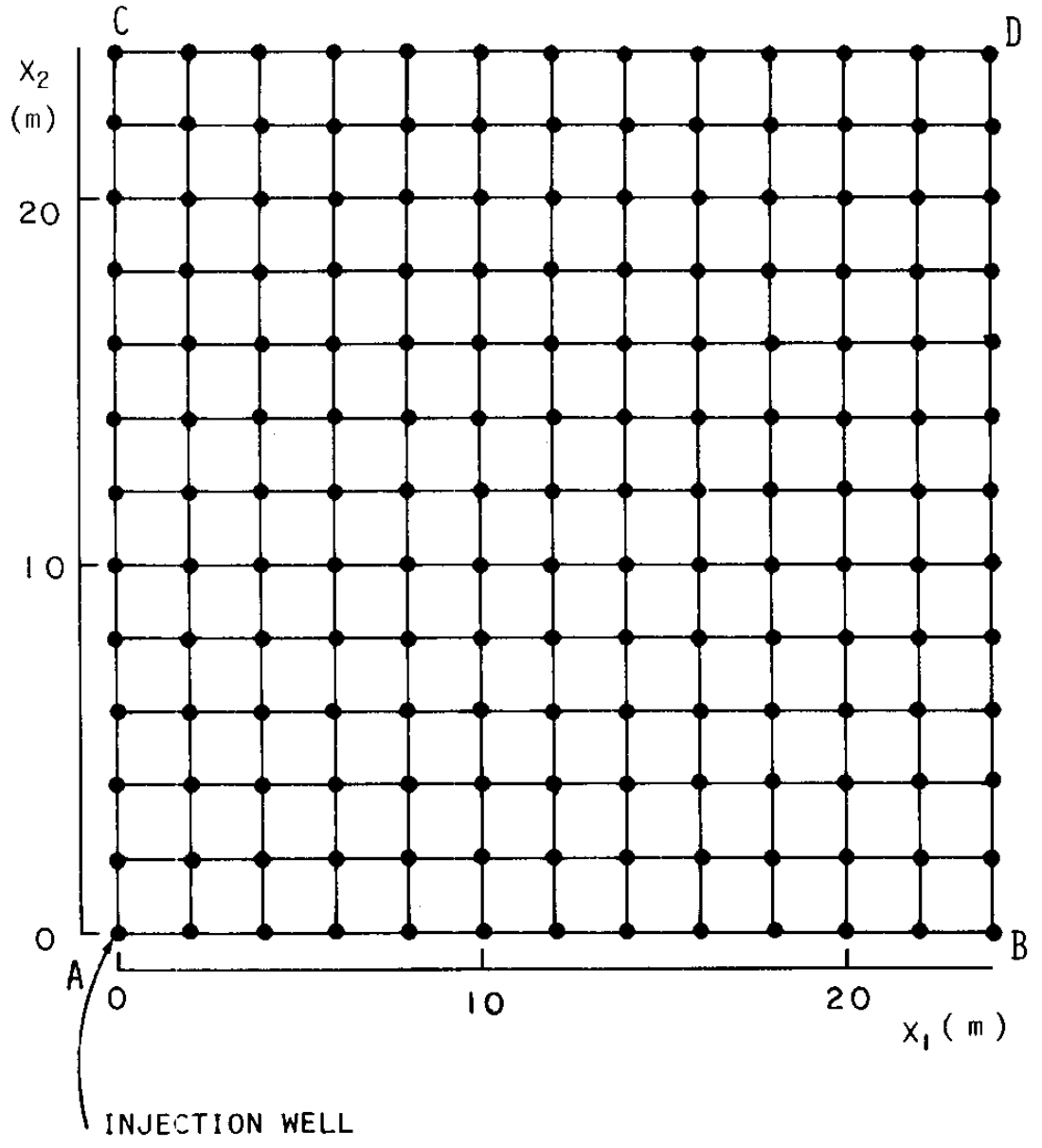


Figure 5.18 - 2-D Finite Element Grid Used in the Injection of Freshwater into a Saline Aquifer Problem

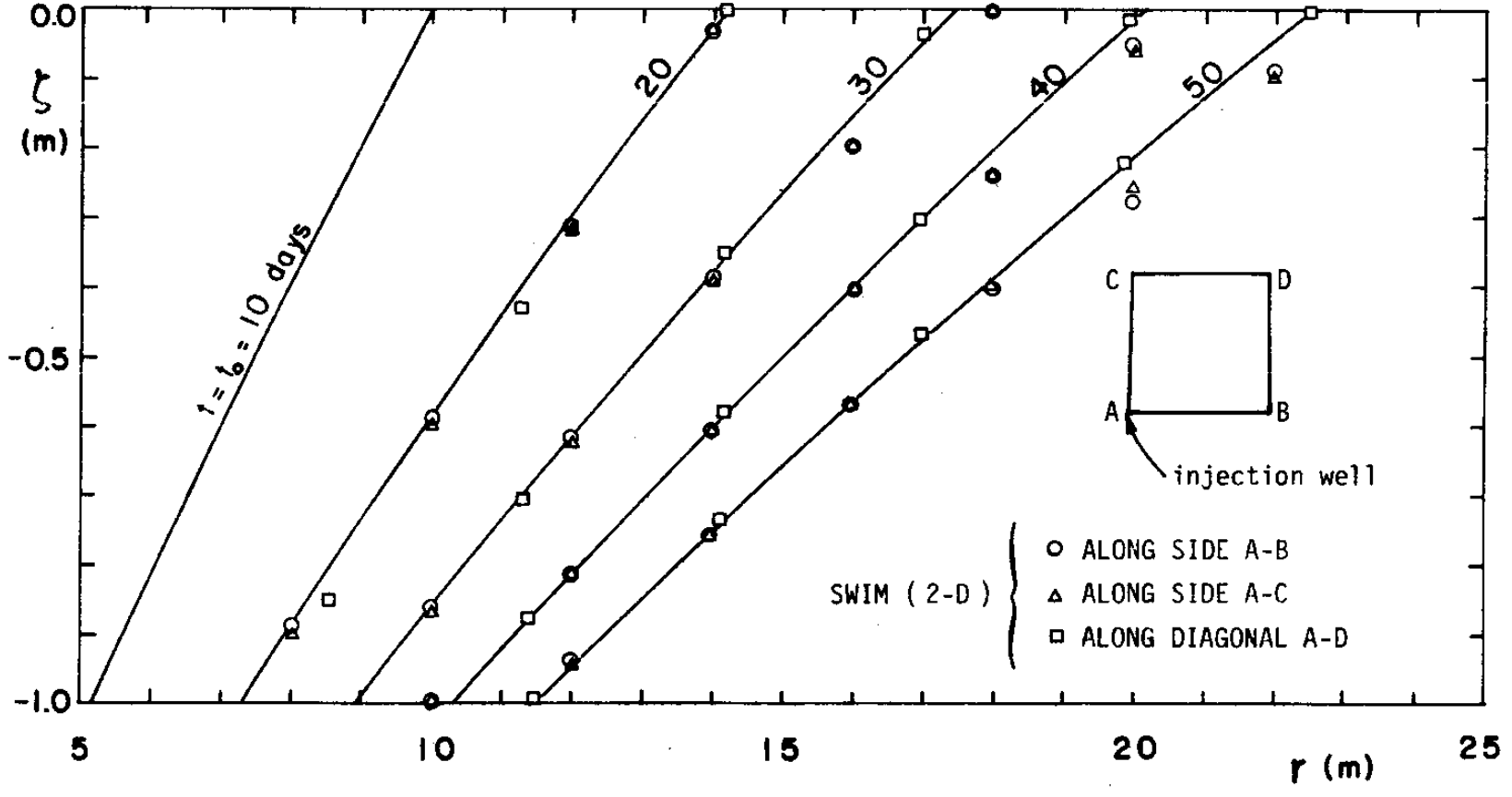


Figure 5.19 - Interface Movement in the Injection of Freshwater into a Saline Aquifer. Analytical Versus SWIM 2-D Grid Solutions

sis that attributed the upper toe acceleration to the proximity of the boundaries.

In this example, SWIM has shown that it can accurately simulate two dimension interfacial phenomena with moving fronts (the upper and lower toes). Even when a relatively sparse grid with no particular orientation to the flow field is employed, the simulation predictions are satisfactory.

5.7 Seawater Intrusion Toward a Coastal Pumping Well

Coastal pumping wells are used for water supply by pumping freshwater, and for control of seawater intrusion, by pumping freshwater and/or seawater. This situation can also be modeled using SWIM. In this section, a single pumping well withdrawing from the freshwater layer is considered, and the position of the toe is determined for transient and steady state conditions.

In Figure 5.20, a schematic representation of the situation modeled is shown. Strack (1976) derived a single-potential steady state solution for the piezometric freshwater head in this type of coastal aquifer with a pumping well, with (Girinskii) potential Φ given by:

$$\Phi = Q_{\infty}^{f} + \frac{Q_w}{2\pi} \ln \left[\frac{(x_1 - x_w)^2 + x_2^2}{(x_1 + x_w)^2 + x_2^2} \right]^{1/2} \qquad (5.5)$$

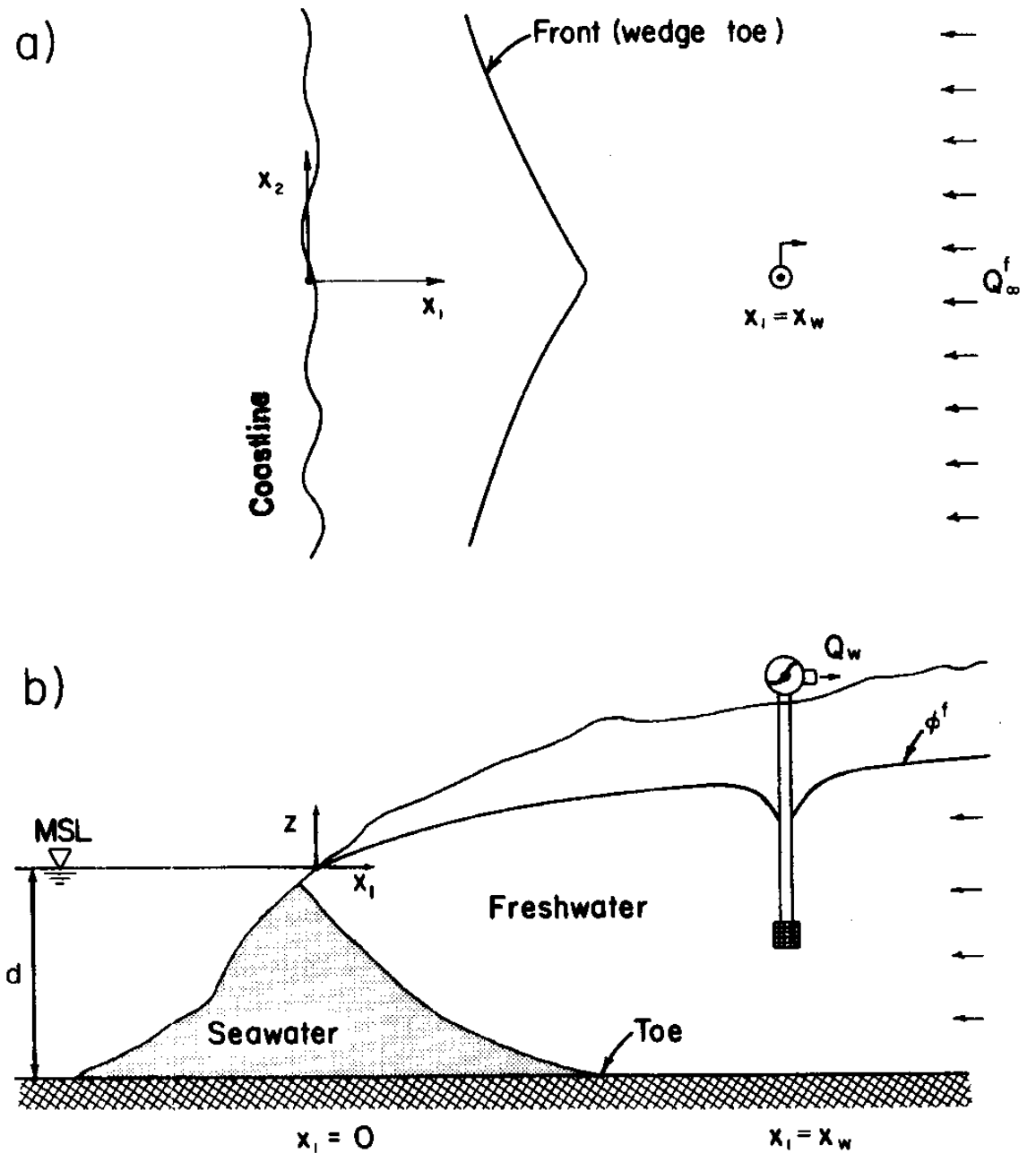


Figure 5.20 - Schematic Representation of Seawater Intrusion Toward a Coastal Pumping Well. a) Plan View; b) Cross Section by the Well.

where:

Q_∞^f = ambient flow per unit width in the negative x direction $[L^2/T]$

Q_w = pumping rate $[L^3/T]$

x_w = distance between the coast line and the well [L]

x_1, x_2 = Cartesian coordinates [L] .

Φ takes on the following values:

above the interface:

$$\Phi = \frac{1}{2} K \frac{\gamma^s}{\Delta\gamma} (\phi^f - d) \tag{5.6}$$

inland of the toe:

$$\Phi = \frac{1}{2} K(\phi^f)^2 - \frac{1}{2} K \frac{\gamma^s}{\gamma^f} d^2 \tag{5.7}$$

where:

$K = K^f$ = permeability in the freshwater zone

d = aquifer depth below the MSL

The following parameter values were used in the SWIM simulations:

Q_o = 1.0 m^2/day (at x_1 = 2000 m, $0 \leq x_2 \leq 2000$ m)

x_w = 600.0 m

K^f = 70.0 m/day

d = 20.0 m

γ^f = 1.0 g/cm^3

γ^s = 1.025 g/cm^3

At the coastline a first type saltwater ($\phi^s = 0$) and third type freshwater coastal boundary condtions were used ($K'_c = K^f = K$). Along the inland boundary a second type specified freshwater flux $Q_x^f = -Q_o$ was used. A "no flow" boundary of symmetry was used for both layers along the x_2 axis ($x_2 = 0$, $0 \leq x_1 \leq 2000$ m). A no flow boundary was also used on the parallel boundary ($x_2 = 2000$ m, $0 \leq x_1 \leq 2000$ m), at a distance that was hopefully far enough from the well so as not to affect the solution employed. Two freshwater pumping rates were tested, $Q_w = 300$ m^3/day and $Q_w = 400$ m^3/day. The finite element grid is shown in Figure 5.21. It has more detail near the well in order to capture locally sharp gradients of piezometric head. The grid has 59 elements and 121 nodes, and only covers half of the space due to the symmetry of the problem. The solution of the transient problem required a lumped storage matrix in order to avoid a non-positive definite effective conductivity matrix (see Appendix B).

The first simulation performed assumed $Q_w = 0$ in order to determine the initial position of the interface, prior to pumping. Then at $t = t_o = 0$ pumping started at a rate of $Q_w = 300$ m^3/day and the transient simulation was performed until a new steady state was reached. The resulting toe positions for the steady states and at certain times during the intervening transient are shown in Figure 5.22 along with the steady state analytical solutions. There is no analytical transient solution with which to compare (unless the Ghyben-Herzberg approximation is invoked). The agreement between the analytical and numerical steady state solutions is very good. The small discrepancy for $x_2 \geq 1000$ m is due to the presence of the boundary at $x_2 = 2000$ m. That is, the numerical solution agrees perfectly with the analytical solution when

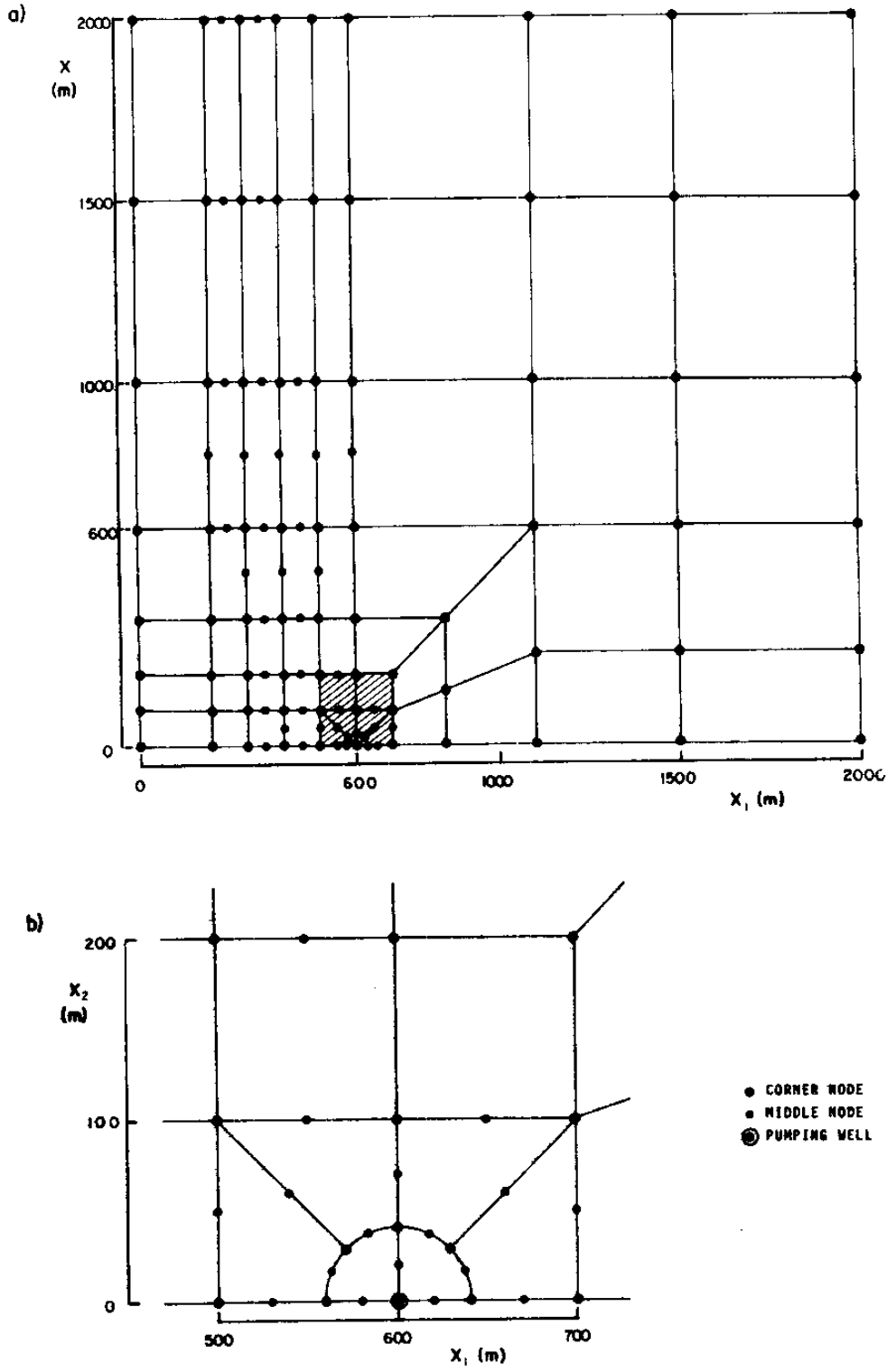


Figure 5.21 -- a) Finite Element Grid Used for Seawater Intrusion Toward a Coastal Pumping Well Problem;
b) Detail of the Shaded Area of a)

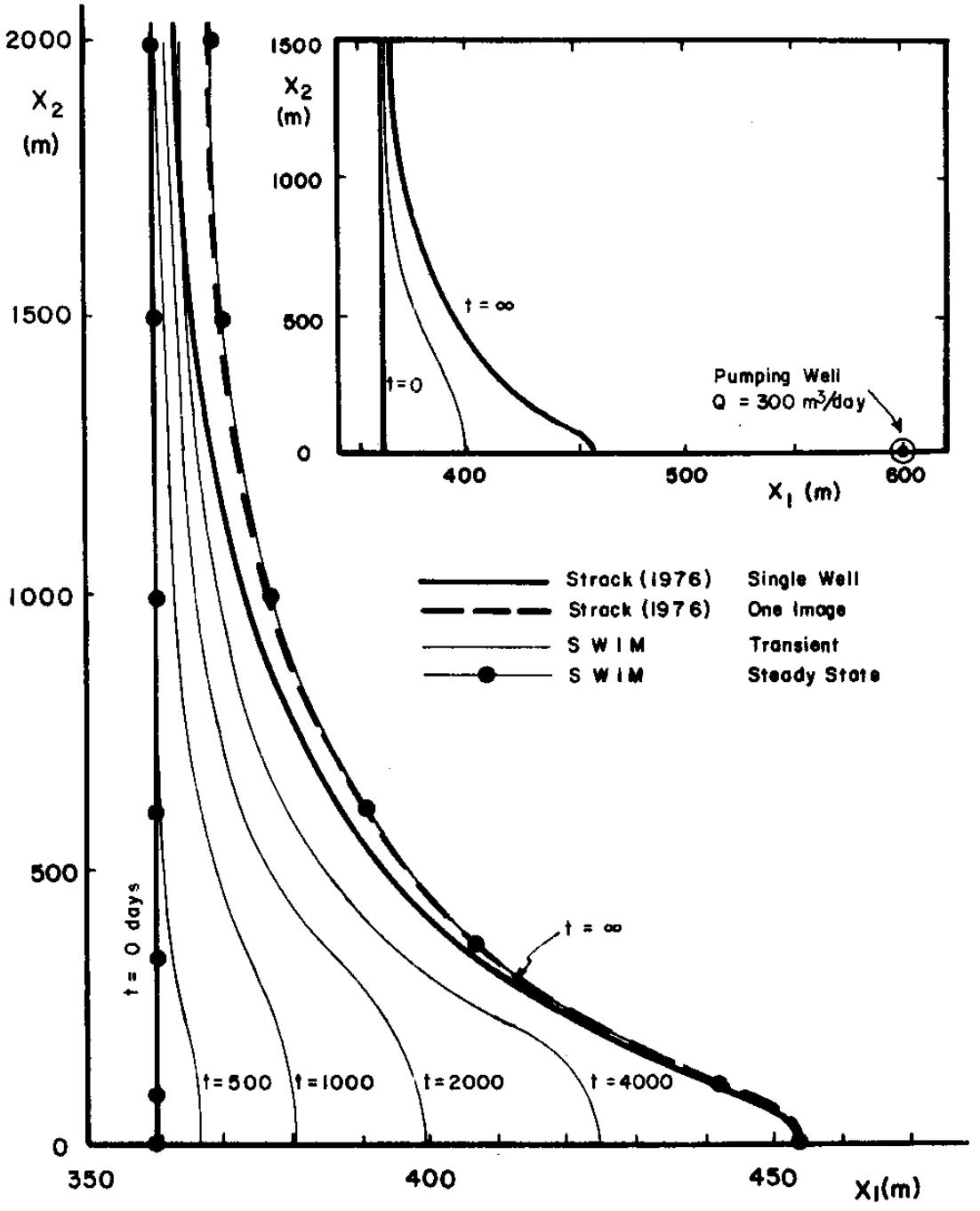


Figure 5.22 - Toe Movement for Seawater Intrusion Toward a Coastal Pumping Well ($Q_w = 300 m^3/day$)

pumping wells are placed parallel to the coast, and 4000 m apart. The first order approximation of this is the use of one image pumping well 2000 m on the other side of the boundary at x_2 = 2000 m: the effect is shown as the dashed line in Figure 5.22. Figure 5.23 presents the analytical and SWIM steady state solutions for the interface profile at different values of x_2, and the piezometric head for x_2 = 0 m. Again, excellent agreement was obtained. The only difference between these solutions is related to the value of the opening to the sea, and corrresponding piezometric heads. The opening has a finite value in the SWIM solution with its third type freshwater coastal boundary condition, and an unrealistic zero value for the analytical solution.

In another set of simulations the pumping rate was increased to Q_w = 400 m^3/day. The steady state solution for this case is presented in Figures 5.24, for the toe location, and 5.25, for the interface and water table profiles. With this new pumping rate the toe advances and some seawater upconing appears under the well. The comments made for the previous test with Q_w = 300 m^3/day are also valid here. The insert of Figure 5.24 shows a more detailed comparison of toe locations for the analytical and SWIM solutions in the upconing area. In this example the toe location for SWIM was computed using the interpolation functions, and the values of the interface elevation at the nodes just seaward and inland of the toe. For the nodes inland of the toe the interface elevation was calculated using computed freshwater and seawater piezometric heads, and Eq. 3.3. The agreement between numerical

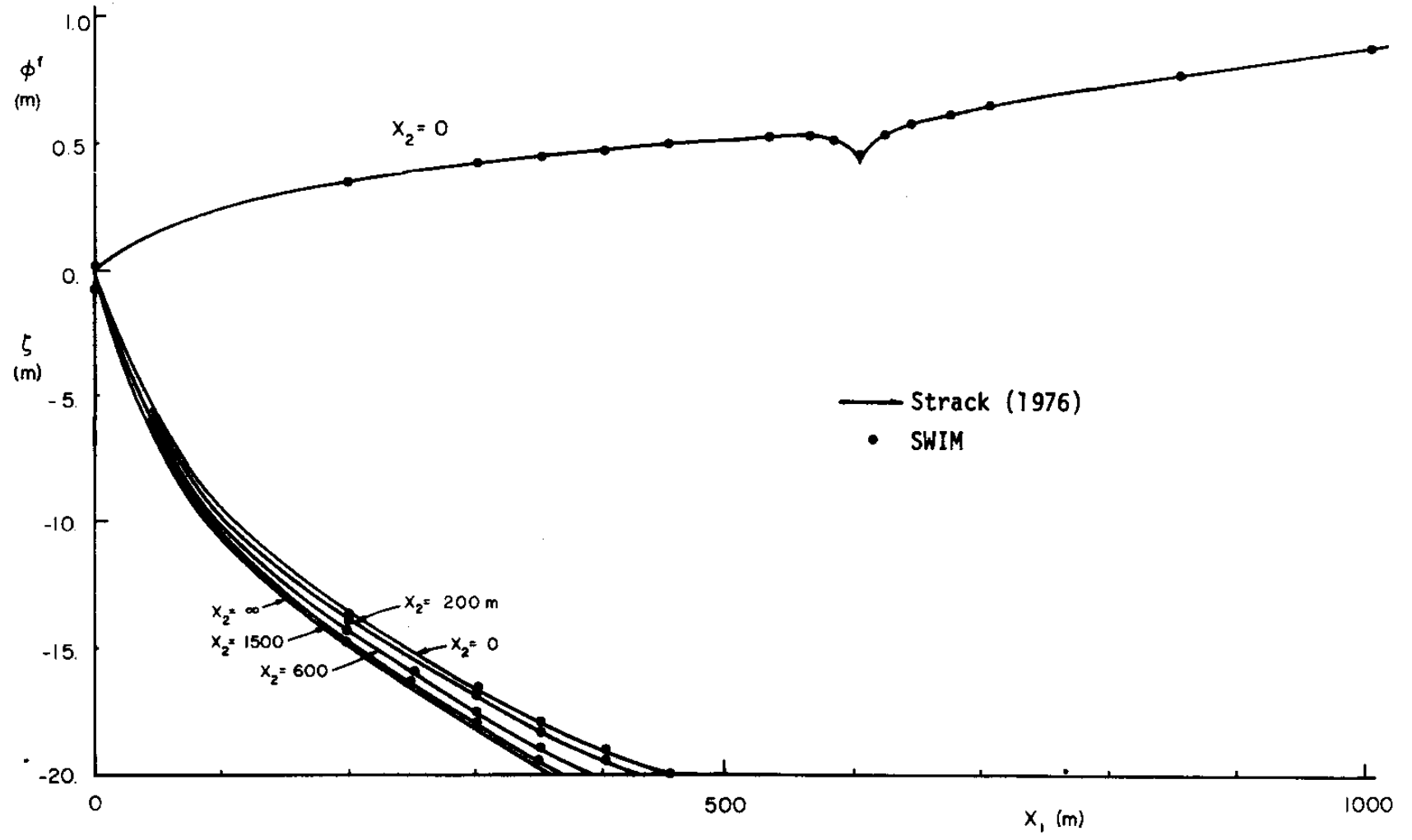


Figure 5.23 - Steady State Position of the Interface (ζ) and Piezometric Head (ϕ^f) for Seawater Intrusion Toward a Coastal Pumping Well ($Q_w = 300 m^3/day$)

NOTE: Different Vertical Scales Are Used for ζ and ϕ^f.

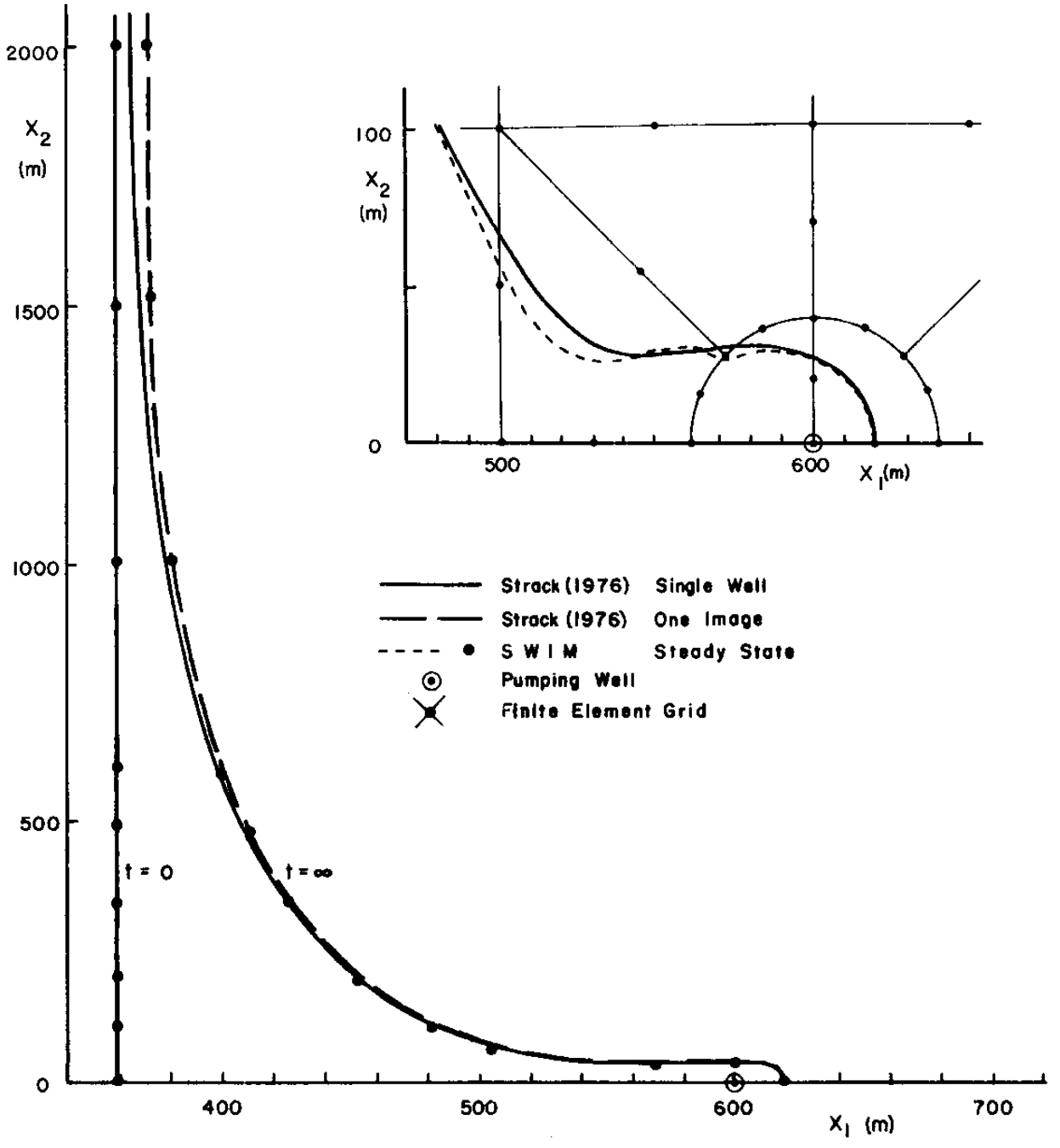


Figure 5.24 - Toe Movement for Seawater Intrusion Toward a Coastal Pumping Well ($Q_w = 400\ m^3/day$)

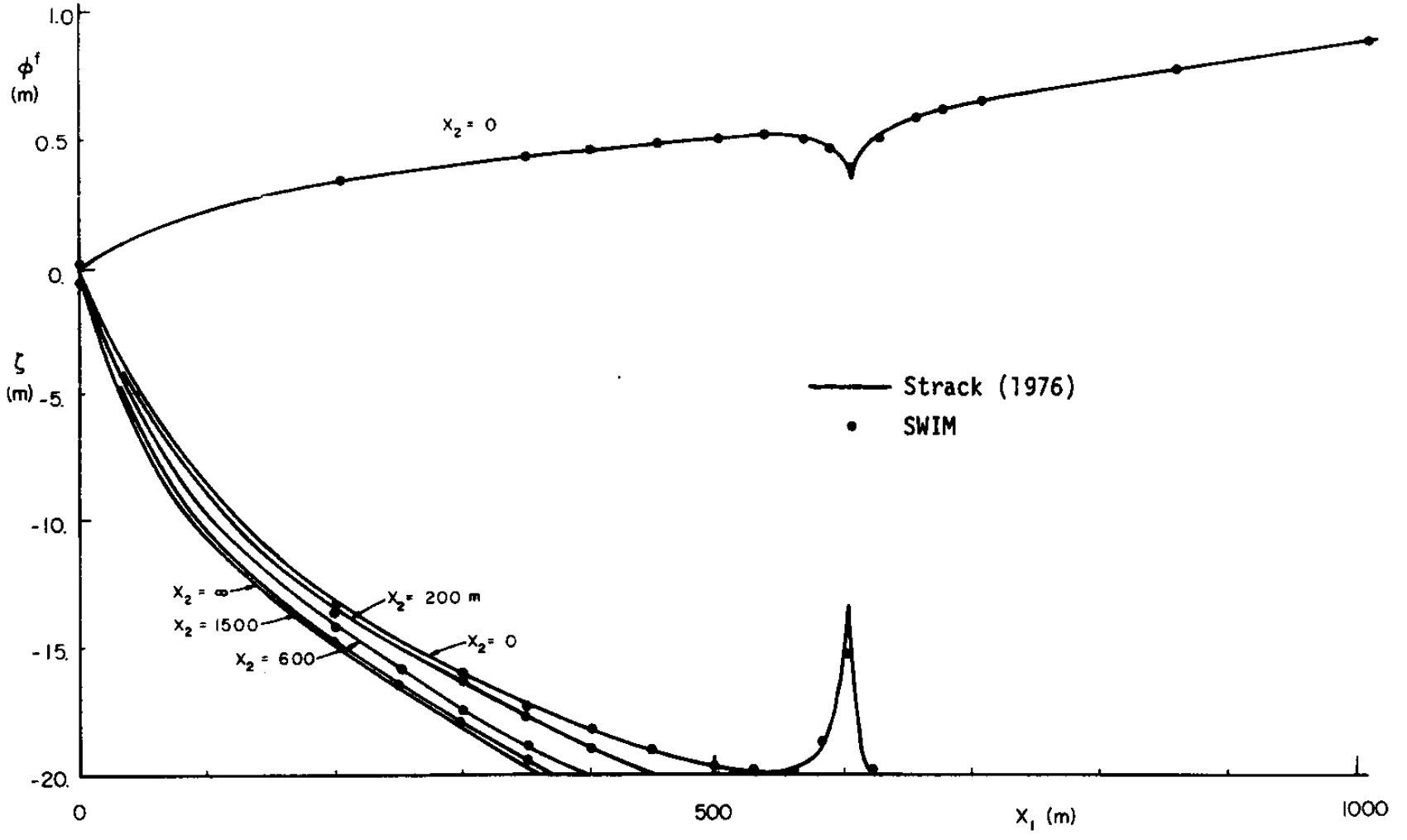


Figure 5.25 - Steady State Position of the Interface (ζ) and Piezometric Head (ϕ^f) for Seawater Intrusion Toward a Coastal Pumping Well ($Q_w = 400\ m^3/day$)

NOTE: Different Vertical Scales Are Used for ζ and ϕ^f.

and analytical solutions is excellent for the type of spatial discretization used.

SWIM shows in these tests that it can be used as a powerful tool to model the effects of pumping activities on seawater intrusion, that it can be used to aid in the design of seawater intrusion barriers employing pumping and/or recharge wells, and that it can simulate seawater upconing toward freshwater pumping wells.

5.8 Summary

In this chapter SWIM was applied to several situations encountered in groundwater hydrology involving two fluids separated by a moving immiscible interface. SWIM solutions were compared with analytical and experimental results, and other numerical simulations. The situations examined were: classical one dimensional seawater intrusion, with both advancing and retreating seawater wedges; development of a freshwater lens over seawater; seawater intrusion in a leaky coastal aquifer; injection of freshwater into a saline aquifer; and seawater intrusion toward a coastal pumping well. These applications demonstrate SWIM's applicability to 1-D and 2-D, steady state or transient flow situations with a variety of boundary conditions, sources and sinks, etc.

Several other issues were investigated, such as the validity of the Ghyben-Herzberg assumption for the transient analysis of the development of a freshwater lens, and the capability of SWIM to represent interface upconing beneath a pumping well.

Chapter 6

SUMMARY AND CONCLUSIONS

6.1 Background

Seawater intrusion along coastlines and under offshore islands is a classical result of groundwater development. On a regional scale vertical effects, such as mixing along the interface between freshwater and seawater, and vertical flow are often negligible, and a two-layer horizontal flow model of the regional flow system is appropriate. Freshwater is the upper layer, seawater is the lower layer, and an immiscible interface between the two is assumed. The Dupuit assumption is used to average vertically all the quantities and parameters. When an immiscible interface is assumed, two situations can be encountered: the interface intersects the bottom of the aquifer defining a toe; or the interface never intersects the bottom of the aquifer, that is, the freshwater body has the shape of a lens floating over the seawater. This report described a numerical model that simulates these situations accurately and effectively.

6.2 Previous Work

The immiscible interface approach to seawater intrusion modeling is much more commonly used than the alternative approach which accounts for dispersion on the mixing zone, mainly because it is easier to implement and calibrate. Existing immiscible interface models have followed one of two approaches:

-the aquifer is very thick, therefore a toe never occurs;

-mesh regeneration is used to track the toe (it's accurate but very expensive to run and besides only a simple 1-D model of this type was found).

The present report introduces a third approach.

6.3 Summary

A finite element model named SWIM, an acronym for Sea Water Intrusion Model, has been developed to model regional flow in coastal aquifers and under offshore islands. A Galerkin finite element statement was used for space discretization, using mixed quadrilateral, isoparametric elements with from 4 to 8 nodes. The aquifer can be homogeneous or non-homogeneous, isotropic or anisotropic, leaky or non-leaky, phreatic or confined. Time varying boundary conditions can also be modeled, and steady state and transient solutions are possible.

In the development of SWIM a special emphasis was put on the algorithm used to track the toe without mesh regeneration. This algorithm uses the Gauss quadrature points and a non-linear variation of the permeability across the element to track the toe. The lower toe as well as the upper toe are tracked in this way. Cases in which a toe does not exist can also be modeled by SWIM.

A sensitivity analysis of SWIM, and especially its toe tracking algorithm, was performed. The effects of space discretization, time step, toe tracking algorithm parameters, number of Gauss points,

number of time steps between updating of effective conductivity matrix, and lumped vs. consistent storage matrix were examined. The 1-D gravitational segregation problem in porous media was used as the basis of this analysis.

To judge the versatility and applicability of SWIM several additional simulations were performed. They include classical 1-D seawater intrusion, development of a freshwater lens over seawater, seawater intrusion in a leaky coastal aquifer, injection of freshwater into a saline aquifer, and seawater intrusion toward a coastal well. The simulations also verified several additional features not examined in the 1-D gravitational segregation problem.

6.4 Conclusions

SWIM has demonstrated that it is an accurate and reasonably efficient model of seawater intrusion in aquifers, whether freshwater occurs as a lens over the seawater, or as a layer over a finite length seawater wedge. In particular:

- -The model accurately represents aquifer situations in which a seawater toe occurs.
- -The toe tracking algorithm has shown a dependence on the time step and space discretization, which affects the accurate representation of the toe movement. A third factor, the extra-layer thickness, also plays a role in this relationship.
- -Local effects, such as interface upconing, are also represented reasonably well by SWIM, especially considering that it is a regional flow model.

-The third type boundary condition at the coastline has performed adequately, in its effort to represent or simulate the non-Dupuit effects occuring in that region.

-For the freshwater lens problem, the SWIM simulation confirmed that the Ghyben-Herzberg approximation is not usually valid for transient problems. Furthermore, the linearizations often used to develop an analytical solution of the transient equations may contribute additional error.

6.5 Future Research

Three features should be implemented in order to make the SWIM code more generally applicable to groundwater problems:

-A time varying storage matrix;

-A time varying third type boundary condition coefficient at the coastline.

-A new third type boundary at pumping wells drawing water from both layers.

The first of these features will make SWIM capable of simulating aquifers changing status from confined to phreatic, or vice-versa. The second of these features may permit a better approximation of the non-Dupuit effects near the coastline during transient flow. The third of these features, in a crude way, will allow the model to simulate the effects of a pumping well that once pumped freshwater only, but later, due to interface upconing, draws fluids from both seawater and freshwater layers.

It can be shown that two phase flow models (say, petroleum reservoir models) can be adapted without revision to the study of seawater intrusion for a lens situation. They can also be used to simulate the toe situation of seawater intrusion by inclusion of a "toe" or front tracking algorithm, of the type described here. A subroutine can be inserted in finite element two phase flow models to check on the presence of a front inside each element. Elements containing a front can be numerically integrated with non-linear permeability and a greater density of Gauss points to evaluate the elemental matrix. Implementing this method should be relatively simple for models that already use Gauss quadrature to integrate over the element area. It would not be very difficult to adopt compact formulas, such as those used in linear triangular elements, to handle the elements containing a front when other schemes of integration are used.

The "toe tracking" algorithm, in one form or another, can also be adapted to other moving boundary problems, particularly the Stefan problem where a few somewhat similar schemes already exist (see Section 2.3.2). Among these is the problem of a change in aquifer status from confined to phreatic, or vice-versa, in which the moving "boundary" separates the two aquifer zones. Current techniques for modeling this groundwater situation are inadequate (see Wilson, Townley and Sá da Costa, 1979).

REFERENCES

Bathe, K.J., "ADINAT - A Finite Element Program for Automatic Dynamic Incremental Nonlinear Analysis of Temperatures", Report 82448-5, Acoustics and Vibrations Laboratory, Mechanical Engineering Department, MIT, May 1977.

Bathe, K.J., "Finite Element Formulation, Modeling and Solution of Nonlinear Dynamic Problems", in Numerical Methods for Partial Differential Equations, ed. by S.V. Parter, Academic Press, New York, 1979.

Bathe, K.J. and M.R. Khoshgoftaar, "Finite Element Free Surface Seepage Analysis Without Mesh Iteration", International J. for Numerical and Analytical Methods in Geomechanics, Vol. 3, 13-22, 1979.

Bathe, K.J. and E.L. Wilson, Numerical Methods in Finite Element Analysis, Prentice-Hall, Inc., New Jersey, 1976.

Bear, J., "The Transition Zone Between Fresh and Salt Waters in Coastal Aquifers," Ph.D. Thesis, University of California at Berkeley, 1960.

Bear, J., "Two Liquid Flows in Porous Media", in Advances in Hydro-Science, Vol. 6, ed. by Ven Te Chow, Academic Press, New York, 1970.

Bear, J., Dynamics of Fluids in Porous Media, American Elsevier Publishing Company, Inc., New York, 1975.

Bear, J., Hydraulics of Groundwater, McGraw-Hill Book Company, New York, 1979.

Bear, J. and G. Dagan, "The Transition Zone Between Fresh and Salt Waters in Coastal Aquifers", Joint Experimental Coastal Ground Water Collectors Project, (Israel), Technical Report No. 4, 1963.

Bear, J. and G. Dagan, "Some Exact Solutions of Interface Problems by Means of the Hodograph Method", J. of Geophysical Research, 6(8), 1563-1572, 1964a.

Bear, J. and G. Dagan, "Moving Interface in Coastal Aquifers", J. of Hydraulics Division of A.S.C.E., 90(HY4), 193-216, 1964(b).

Cheng, R.T. and M.H. Hu, "Study of Fluid Movements Through Causeway", J. of Hydraulics Division of A.S.C.E., 101(HY1), 155-165, 1975.

Collins, M.A. and L.W. Gelhar, "Seawater Intrusion in Layered Aquifers", Water Resources Research, 7(4), 971-979, 1971.

Collins, M.A. and L.A. Gelhar, "Comments on 'The Shape of the Interface in Steady Flow in Stratified Aquifer', by Y. Mualem and J. Bear", Water Resources Research, 13(2), 468-488, 1977.

Collins, M.A., L.A. Gelhar and J.L. Wilson, "Hele-Shaw Model of Long Island Aquifer System", J. of Hydraulics Division of A.S.C.E., 98(HY9), 1701-1714, 1972.

Connor, J.J. and C.A. Brebbia, Finite Element Techniques for Fluid Flow, Newnes-Butterworths, London, 1976.

Crank, J., "Finite Differences Methods," in Moving Boundary Problems in Heat Flow and Diffusion, ed. by J.R. Ockender and W.R. Hodgkins, Clarendon Press, Oxford, 1975.

Crichlow, H.B., Modern Reservoir Engineering - A Simulation Approach, Prentice-Hall, Inc., New Jersey, 1977.

Dalen, V., "Immiscible Flow by Finite Elements", presented at the International Conference on Finite Elements in Water Resources, Princeton, 1976.

Desai, C.S. and J.F. Abel, Introduction to the Finite Element Method-A Numerical Method for Engineering Analysis, Van Nostrand Reinhold Company, New York, 1972.

Desai, C.S. and D.N. Contractor, "Finite Element Analysis of Flow, Diffusion and Salt Water Intrusion in Porous Media", in Formulation and Computational Algorithms in Finite Element Analysis, ed. by K.J. Bathe et al., MIT Press, 958-983, 1977.

Drabe, J. and W. Badon Ghyben, "Nota in Verband de Voorgenomen Putboring Nabij Amsterdam", (Nota in the Probable Results of the Proposed Well Drilling Near Amsterdam), Tijdschrfit van het Koninklijk Institut van Inginieurs, 8-22, The Hague, 1889.

Fetter, C.W., Jr., "Position of the Saline Water Interface Beneath Oceanic Islands", Water Resources Research, 8(5), 1307-1315, 1972.

Fisher, I. and I.C. Medland, "The Multidimensional Stefan Problem: A Finite Element Approach", in Finite Element in Fluids, Vol. 1, ed. by R.H. Gallagher et al., John Wiley and Sons, U.K., 1974.

Gelhar, L.W., J.L. Wilson and J.S. Miller, "Gravitational and Dispersive Mixing in Aquifers", J. of the Hydraulics Division of A.S.C.E., (HY12), 2135-2153, 1972.

Gelhar, L.W., J.L. Wilson, J.S. Miller and J.M. Hamrick, "Density Induced Mixing in Confined Aquifers", Report No. 145, R.M. Parsons Laboratory for Water Resources and Hydrodynamics, MIT, March 1972(a).

Glover, R.E., "The Pattern of Freshwater Flow in a Coastal Aquifer", J. of Geophysical Research, 64(4), 457-459, 1959.

Hantush, M.S., "Unsteady Movement of Fresh Water in Thick Unconfined Saline Aquifers", Bulletin of International Association of Scientific Hydrology, XII(2), 40-60, 1968.

Hashish, M.A., M.E. Rasmy and A.M. Amer, "One Dimensional Steady State Sea Water Intrusion in Leaky Aquifers", presented in the Conference on Water Resources Planning in Egypt, Cairo, June 1979.

Henry, H.R., "Interfaces Between Salt Water and Fresh Water in Coastal Aquifers", in Sea Water in Coastal Aquifers,ed. by H.H. Cooper et al., Geological Survey Water-Supply Paper 1613-C, Washington, D.C., 35-69, 1964a.

Henry, H.R., "Effects of Dispersion on Salt Encroachment in Coastal Aquifers", in Sea Water in Coastal Aquifers, ed. by H.H. Cooper et al., Geological Survey Water-Supply Paper 1613-C, Washington, D.C., 70-83, 1964b.

Herzberg, A., "Die Wasserversorgung Einiger Nordsee Boder" (The Water Supply on Parts of the North Sea Coast), J. Gasbebeuchtung und Wasserversorgung, Vol. 44, 815-819, 842-844, Munich, 1901.

Hubbert, M.K., "The Theory of Ground Water Motion", J. of Geology 48(8), 785-944, 1940.

Hueber, K.H., The Finite Element Method for Engineers, John Wiley & Sons, New York, 1975.

Intercomp Resource Development and Engineering, Inc., "A Model for Calculating Effects of Liquid Waste Disposal in Deep Saline Aquifers", prepared for U.S.G.S., NTIS PB256-203, June 1976.

Kashef, A.I., "Management of Retardation of Salt Water Intrusion in Coastal Aquifers", Office of Water Research and Technology, Washington, D.C., 1975.

Kashef, A.I., "Control of Salt-Water Intrusion by Recharge Wells", J. of the Irrigation and Drainage Division of A.S.C.E., 102 (IR4), 445-457, 1976.

Kashef, A.I. and J.C. Smith, "Expansion of Salt-Water Zone Due to Well Discharge", Water Resources Bulletin, J. of AWRA, 11(6), 1107-1120, 1975.

Keulegan, G.H., "Ninth Progress Report on Model Laws for Density Currents; An Example of Density Current Flow in Permeable Media", National Bureau of Standards Report No. 3411, Washington, D.C., 1954.

Kishi, Y. and Y. Fukuo, "Studies on Salinization of Groundwater, I.-Theoretical Considerations on the Three-Dimensional Movement of Salt Water Interface Caused by Pumpage of a Confined Groundwater in Fanshpaed Alluvium", J. of Hydrology, 35(1/2), 1-29, 1977.

Kono, I., "Analysis of Interface Problems in Groundwater Flow by Finite Element Method," in Finite Elements in Fluids, Vol. 1, ed. by R.H. Gallagher et al., John Wiley and Sons, U.K., 1974.

Kushner, A.S. and W.H. Walston, Jr., "Conduction and Natural Convection Heat Transfer in a Phase Change Region", in Computational Techniques for Interface Problems, ed. by K.C. Park and D.K. Gartling, A.S.C.E., AMD Vol. 30, 1978.

Lu, C. and R.T. Cheng, "On Seawater Encroachment in Coastal Aquifers", Water Resources Research, 10(3), 1039-1043, 1974.

Lefebvre du Prey, E.J. and L.M. Weill, "Front Displacement in a Fractured Reservoir", in Finite Element Methods in Flow Problems, ed. by J.T. Oden et al., UAH Press, Alabama, 1974.

Lewis, R.W., I.R. White and W.L. Wood, "A Starting Algorithm for the Numerical Simulation of Two-Phase Flow Problems", International Journal for Numerical Methods in Engineering, 12(2), 319-329, 1978.

Liu, P.L.F. and J.A. Liggett, "An Efficient Numerical Method of Two-Dimensional Steady Groundwater Problems", Water Resources Research, 14(3), 385-390, 1978.

McMichael, C.L. and G.W. Thomas, "Reservoir Simulation by Galerkin's Method", Society of Petroleum Engineers Journal, 13(3), 125-138, 1973.

Mercer, J.W. and C.R. Faust, "The Application of Finite-Element Techniques to Immiscible Flow in Porous Media", presented at the International Conference on Finite Elements in Water Resources, Princeton, 1976.

Mualem, Y. and J. Bear, "The Shape of the Interface in Steady Flow in a Stratified Aquifer", Water Resources Research, 10(6), 1207-1215, 1974.

Peaceman, D.W., Fundamentals of Numerical Reservoir Simulation, Elsevier Scientific Publishing Co., The Netherlands, 1977.

Pinder, G.F. and H.H. Cooper, "A Numerical Technique for Calculating the Transient Position of the Saltwater Front", Water Resources Research, 6(3), 875-882, 1970.

Pinder, G.F. and W.G. Gray, Finite Element Simulation in Surface and Subsurface Hydrology, Academic Press, New York, 1977.

Pinder, G.F. and R.H. Page, "Finite Element Simulation of Salt Water Intrusion on the South Fork of Long Island", presented at the International Conference in Finite Elements in Water Resources Princeton, 1976.

Price, H.S., J.C. Cavendish and R.S. Varga, "Numerical Methods of Higher-Order Accuracy for Diffusion-Convection Problems", Society of Petroleum Engineers Journal, 8(3), 293-303, 1968.

Rofail, N., "A Mathematical Model of Stratified Groundwater Flow", Hydrological Sciences-Bulletin, XII(4), 503-512, 1977.

Rumer, R.R. and D.R.F. Harleman, "Intruded Salt-Water Wedge in Porous Media,", J. of Hydraulics Division of the A.S.C.E., 89(HY6), 193-220, 1963.

Rumer, R.R. and J.C. Shiau, "Saltwater Interface in a Layered Coastal Aquifer", Water Resources Research, 4(6), 1235-1247, 1968.

Sá da Costa, A.A.G., "A Two-Dimensional Finite Element Groundwater Flow Model", M.Sc. Thesis, MIT, January 1978.

Sá da Costa, A.A.G. and J.L. Wilson, "Comment on 'A Mathematical Model of Stratified Groundwater Flow' by N. Rofail", Hydrological Sciences-Bulletin, XXIII(2), 267-268, 1978.

Segol, G. and G.F. Pinder, "Transient Simulation of Saltwater Intrusion in Southeastern Florida", Water Resources Research, 12(1), 65-70, 1976.

Segol, G., G.F. Pinder and W.G. Gray, "A Galerkin-Finite Element Technique for Calculating the Transient Position of the Saltwater Front", Water Resources Research, 11(2), 343-347, 1975.

Shamir, U. and G. Dagan, "Motion of the Seawater Interface in Coastal Aquifers: A Numerical Solution", Water Resources Research, 7(3), 644-657, 1971.

Spivak, A., H.S. Price and A. Settari, "Solution of the Equations for Multidimensional Two-Phase, Immiscible Flow by Variational Methods", Scoiety of Petroleum Engineers Journal, 17(1), 27-41, 1977.

Strack, O.D.L., "Single-Potential Solution for Regional Interface Problems in Coastal Aquifers", Water Resources Research, 12(6), 1165-1174, 1976.

Todd, D.K. and L. Huisman, "Groundwater Flow in the Netherland Coastal Dunes", J. of Hydraulics Division of A.S.C.E., 85(HY7), 63-81, 1959.

Van Dam, J.C., "Partial Depletion of Saline Groundwater by Seepage", J. of Hydrology, 29(3/4), 315-339, 1976.

Van der Veer, P., "Analytical Solution for Steady Interface Flow in a Coastal Aquifer Involving a Phreatic Surface with Precipitation", J. of Hydrology, 34(1/2), 1-11, 1977.

Verou, M., "Sea Water Intrusion in a Leaky Coastal Aquifer", M.Sc. Thesis, MIT, January 1978.

Verruijt, A., "A Note on the Ghyben-Herzberg Formula", Bulletin of the International Association of Scientific Hydrology, 13(4), 43-46, 1968.

Wilson, J.L., M.E. Rasmy and M.A. Hashish, "A Finite Element Model for the Nile Delta Aquifer", presented in the Conference on Water Resources Planning in Egypt, Cairo, June 1979.

Wilson, J.L., L. Townley and A. Sá da Costa. "Mathematical Development and Verification of a Finite Element Aquifer Flow Model AQUIFEM-1" Tecnology Adaptation Program Report 79-2, MIT, 1979.

Zienkiewicz, O.C. The Finite Element Method, 3rd edition, Mcgraw-Hill Book Company(U.K.), London, 1977.

Appendix A

DERIVATION OF THE GOVERNING EQUATIONS FOR SEAWATER INTRUSION IN AQUIFERS

A.1 Introduction

The objective of this Appendix is the derivation of the general governing equations for groundwater flow in coastal or island aquifers experiencing seawater intrusion. The two equations obtained, one for the freshwater phase, the other for the seawater phase, will be used in the development of the two dimensional numerical model using the finite element method, SWIM.

The general situation involves freshwater flowing toward the sea due to a gradient, with the denser seawater forming a wedge underneath. Figure A-1 illustrates some of these typical situations. It is important to realize that these figures are highly distorted in the vertical scale.

Freshwater and seawater are miscible fluids, and a transition zone exists between them, in which the salt concentration and thus the density of water increases toward the sea. This zone is created by the flow field and the effects of hydrodynamic dispersion. In many circumstances, its width is small when compared with the aquifer thickness. To neglect the transition zone is equivalent to assuming that an immiscible interface separates the regions occupied by the two fluids. This type of assumption is acceptable in many cases of practical interest, and greatly simplifies the solution of the problem.

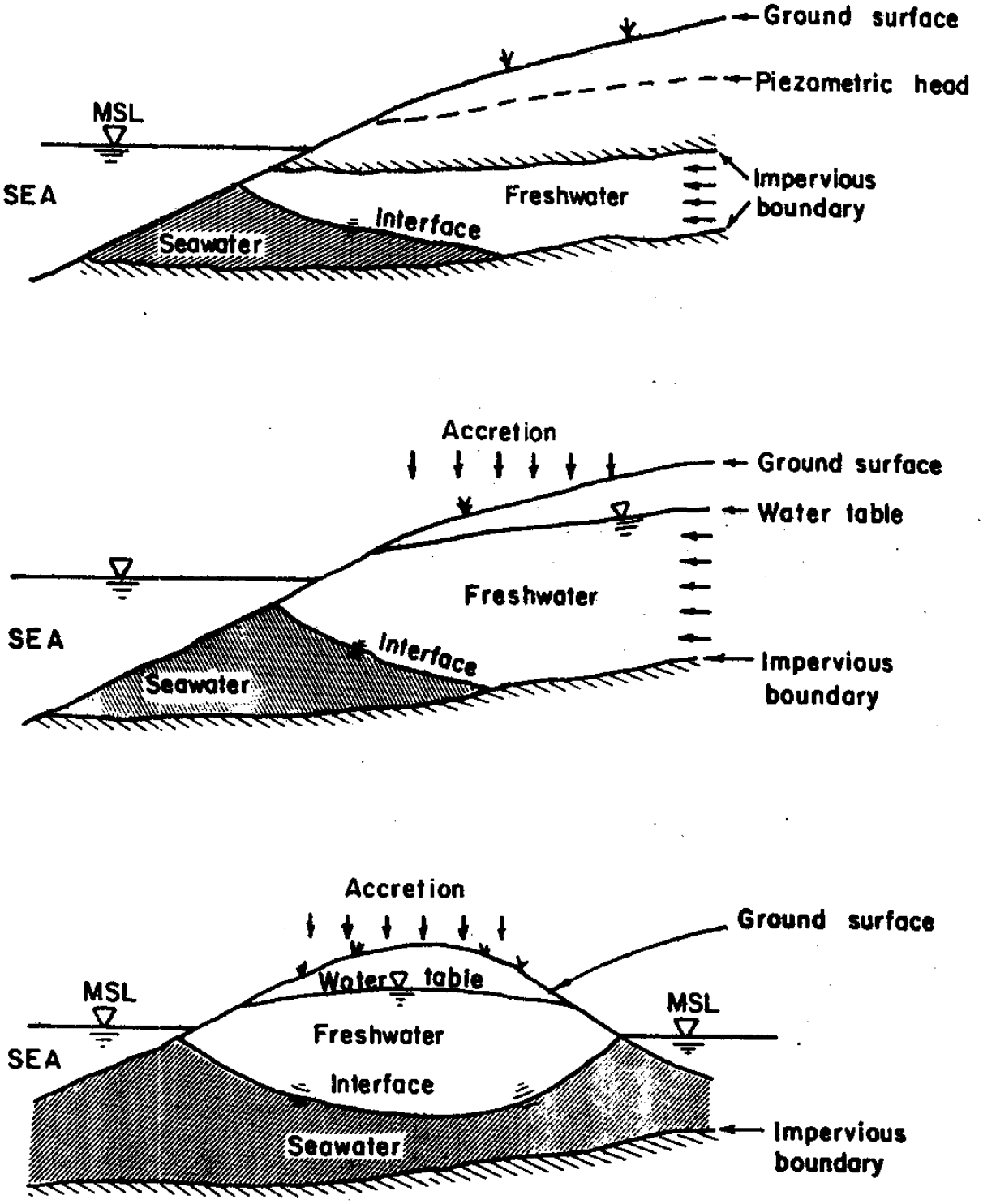


Figure A.1 - Sewater Intrusion in Different Types of Aquifers

A.2 Derivation of the Governing Equations

A.2.1 Confined Aquifers

In the derivation of the governing equations, Darcy's Law is assumed valid:

$$\underline{q} = -\underline{\underline{k}} \nabla \phi$$

where $\underline{q}$ = specific discharge vector with components

$$q_x = -K_x \frac{\partial \phi}{\partial x}$$

$$q_y = -K_y \frac{\partial \phi}{\partial y}$$

$$q_z = -K_z \frac{\partial \phi}{\partial z}$$

$\underline{\underline{k}}$ = permeability tensor (for sake of simplicity in the derivation only--principal directions are assumed)

ϕ = piezometric head

∇ = grad.

The underline $\_$ represents a vector and $\underline{\underline{\ }}$ a matrix (tensor) quantity.

For a confined aquifer the general conservation of mass equation in three dimensions is:

$$\frac{\partial}{\partial t} (n\rho) = \text{div}(\rho q)$$

or in Cartesian coordinates and using Darcy's Law as given above the expression becomes (Bear, 1972):

$$\frac{\partial}{\partial x}\left(K_x \frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial \phi}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_z \frac{\partial \phi}{\partial z}\right) = S_s \frac{\partial \phi}{\partial t} \qquad (A.1)$$

where $S_s = (n\beta + \alpha)\gamma$ or $[n\beta + (1-n)\alpha]\gamma$ = specific storage,

n = effective porosity,

β = water compressibility

α = soil vertical compressibility

γ = water specific weight.

The choice of the form of specific storage depends on the derivation taken (Bear, 1979).

Since a two-dimensional equation in the horizontal plane is sought, Eq. A.1 must be integrated in the vertical direction, z, between the confining boundaries, $z_1(x,y)$ and $z_2(x,y)$. Using Liebnitz' rule:

$$\frac{\partial}{\partial x}\left[\int_{B(x)}^{A(x)} f(x,w)dw\right] = \int_{B(x)}^{A(x)} \left[\frac{\partial f(z,w)}{\partial x}\right] dw + f(x,A)\frac{\partial A(x)}{\partial x} - f(x,B)\frac{\partial B(x)}{\partial x}$$

and neglecting the variation of S_s with z*, the result is:

$$\frac{\partial}{\partial x}\left(\int_{z_2}^{z_1} K_x \frac{\partial \phi}{\partial x}\, dz\right) + \frac{\partial}{\partial y}\left(\int_{z_2}^{z_1} K_y \frac{\partial \phi}{\partial y}\, dz\right) - K_x \frac{\partial \phi}{\partial x}\Big|_{z_1} \frac{\partial z_1}{\partial x}$$

$$- K_y \frac{\partial \phi}{\partial y}\Big|_{z_1} \frac{\partial z_1}{\partial y} + K_x \frac{\partial \phi}{\partial x}\Big|_{z_2} \frac{\partial z_2}{\partial x} + K_y \frac{\partial \phi}{\partial y}\Big|_{z_2} \frac{\partial z_2}{\partial y} + K_z \frac{\partial \phi}{\partial z}\Big|_{z_1}$$

$$- K_z \frac{\partial \phi}{\partial z}\Big|_{z_2} = S_s \frac{\partial}{\partial t}\left(\int_{z_2}^{z_1} \phi dz\right) - S_s \phi\Big|_{z_1} \frac{\partial z_1}{\partial t} + S_s \phi\Big|_{z_2} \frac{\partial z_2}{\partial t} \qquad (A.2)$$

---

*If there is a significant variation of S_s with z, an averaged value can be used, or the aquifer system can be divided into several layers such that a constant S_s within each layer can be assumed without introducing significant error. One can also exactly account for this variation by introducing additional terms in the final equation, which generally can be neglected.

Eq. A.2 can be simplified using the following notation:

$$\underline{q}' = \begin{bmatrix} q_x \\ q_y \end{bmatrix} \quad \text{where } \underline{q} = \underline{q}' + \underline{q}_z$$

$$\nabla' = \text{grad}_{x,y} \quad \text{where } \nabla = \nabla' + \frac{\partial}{\partial z}\underline{\ell}_z$$

$$\underline{\ell}_z = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \text{direction cosine of the vertical direction.}$$

Introducing these definitions in Eq. A.2 the result is:

$$\nabla' \int_{z_2}^{z_1} \underline{q}' \, dz - \underline{q}'\Big|_{z_1} \nabla' z_1 + \underline{q}'\Big|_{z_2} \nabla' z_2 - \underline{q}_z\Big|_{z_1} + \underline{q}_z\Big|_{z_2}$$

$$= S_s \frac{\partial}{\partial t} \left(\int_{z_2}^{z_1} \phi dz\right) - S_s \phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial t} + S_s \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial t}$$

Adding the terms representing the fluxes at the confining boundaries leads to:

$$\nabla' \int_{z_2}^{z_1} \underline{q}' \, dz - \underline{q}\Big|_{z_1} \cdot \nabla(z-z_1) + \underline{q}\Big|_{z_2} \cdot \nabla(z-z_2)$$

$$= S_s \frac{\partial}{\partial t} \left(\int_{z_2}^{z_1} \phi \, dt\right) - S_s \phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial t} + S_s \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial t} \qquad \text{(A.3)}$$

The definition of flow normal to a surface s is given by $q_n = \underline{q}\,\underline{n}$, where $\underline{n} = \nabla s$ is the unit vector normal to the surface. The equations defining the confining boundaries are:

top boundary: $s_1 = z - z_1 = 0;$

bottom boundary: $s_2 = z - z_2 = 0.$

If these boundaries are considered impervious:

$$q_n\Big|_{z_1} = \underline{q}\Big|_{z_1} \cdot \nabla(z-z_1) = 0$$

$$q_n\Big|_{z_2} = \underline{q}\Big|_{z_2} \cdot \nabla(z-z_2) = 0$$

then Eq.A.3 is simplified:

$$\nabla' \int_{z_2}^{z_1} \underline{q}' dz = S_s \frac{\partial}{\partial t} \left(\int_{z_2}^{z_1} \phi dz \right) - S_s \phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial t} + S_s \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial t}$$

In Cartesian notation this equation becomes:

$$\frac{\partial}{\partial x} \left(\int_{z_2}^{z_1} K_x \frac{\partial \phi}{\partial x} dz \right) + \frac{\partial}{\partial y} \left(\int_{z_2}^{z_1} K_y \frac{\partial \phi}{\partial y} \right) dz) = S_s \frac{\partial}{\partial t} \left(\int_{z_2}^{z_1} \phi dz \right)$$

$$- S_s \phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial t} + S_s \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial t} \qquad (A.4)$$

Using Liebnitz' rule again to carry out the integration on the left hand side of Eq. A.4, and neglecting variations of K_x and K_y with z (the same assumption made for S_s in obtaining Eq. A.2) gives

$$\frac{\partial}{\partial x} \left[K_x \frac{\partial}{\partial x} \left(\int_{z_2}^{z_1} \phi dz \right) \right] + \frac{\partial}{\partial x} \left[K_x \left(-\phi\Big|_{z_1} \frac{\partial z_1}{\partial x} + \phi\Big|_{z_2} \frac{\partial z_2}{\partial x} \right) \right] +$$

$$+ \frac{\partial}{\partial y} K_y \frac{\partial}{\partial y} \left(\int_{z_2}^{z_1} \phi dz\right) + \frac{\partial}{\partial y} K_y \left(-\phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial y} + \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial y}\right)$$

$$= S_s \frac{\partial}{\partial t} \left(\int_{z_2}^{z_1} \phi dz\right) - S_s \phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial t} + S_s \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial t}$$

Defining average piezometric head $\bar{\phi}$ and aquifer thickness b,

$$\bar{\phi} = \frac{1}{b} \int_{z_2}^{z_1} \phi dz \qquad \text{where } b = z_1 - z_2$$

and carrying out the derivatives of $b\bar{\phi}$, one arrives at:

$$\frac{\partial}{\partial x} \left(K_x b \frac{\partial \bar{\phi}}{\partial x}\right) + \frac{\partial}{\partial x} \left[K_x \left(\bar{\phi} \frac{\partial (z_1 - z_2)}{\partial x} - \phi\Big|_{z_1} \frac{\partial z_1}{\partial x} + \phi\Big|_{z_2} \frac{\partial z_2}{\partial x}\right)\right] +$$

$$+ \frac{\partial}{\partial y} \left(K_y b \frac{\partial \bar{\phi}}{\partial y}\right) + \frac{\partial}{\partial y} \left[K_y \left(\bar{\phi} \frac{\partial (z_1 - z_2)}{\partial y} - \phi\Big|_{z_1} \frac{\partial z_1}{\partial y} + \phi\Big|_{z_2} \frac{\partial z_2}{\partial y}\right)\right] =$$

$$= S_s b \frac{\partial \bar{\phi}}{\partial t} + S_s \left(\bar{\phi} \frac{\partial (z_1 - z_2)}{\partial t} - \phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial t} + \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial t}\right) \qquad \text{(A.5)}$$

In confined aquifers one can often assume essentially horizontal flow, which is equivalent to assuming that the equipotential lines are vertical. Therefore, along a vertical line, $\bar{\phi} \simeq \phi\Big|_{z_1} \simeq \phi\Big|_{z_2} \simeq \phi$, and Eq. A.5 reduces to:

$$\frac{\partial}{\partial x} \left(K_x b \frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y} \left(K_y b \frac{\partial \phi}{\partial y}\right) = S_s b \frac{\partial \phi}{\partial t} \qquad \text{(A.6)}$$

which is the governing equation for a confined flow between two impervious layers.

The assumption of essentially horizontal flow in an aquifer with seawater intrusion can introduce appreciable error, especially near the zone of the freshwater opening to the sea, where the flow is no longer horizontal and the equipotentials are no longer vertical. However, the simplifications introduced by such as assumption justify its adoption for many field problems. A detailed analysis of the flow field in the zone near the coast or in areas with significant vertical recharge should be performed in each particular case, in order to verify the adequacy of this approximate approach.

At this point assume that the lower confining layer represents the freshwater/seawater interface $\xi(x,y,z,t) = z - \zeta = 0$, where ζ is the interface elevation (see Fig. A.2). The kinematic boundary condition for this surface,

$$\frac{d\xi}{dt} = \frac{\partial \xi}{\partial t} + \frac{q}{n} \nabla \xi = 0 \qquad (A.7)$$

is introduced in Eq. A.3, which is now written as:

$$\nabla' \int_{\zeta}^{z_1} \underline{q}' dz - \underline{q}\Big|_{z_1} \cdot \nabla(z-z_1) + \underline{q}\Big|_{\zeta} \cdot \nabla(z-\zeta) =$$

$$= S_s \frac{\partial}{\partial t} \left(\int_{\zeta}^{z_1} \phi dz\right) - S_s \phi\Big|_{z_1} \cdot \frac{\partial z_1}{\partial t} + S_s \phi\Big|_{\zeta} \cdot \frac{\partial \zeta}{\partial t} \qquad (A.8)$$

Introducing Eq. A.7 in Eq. A.8, and using the Liebnitz rule and the same procedure and assumptions used to obtain Eq. A.6, leads to

$$\frac{\partial}{\partial x}\left(K_x b \frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y b \frac{\partial \phi}{\partial y}\right) = S_s b \frac{\partial \phi}{\partial t} - n \frac{\partial \zeta}{\partial t} \qquad (A.9)$$

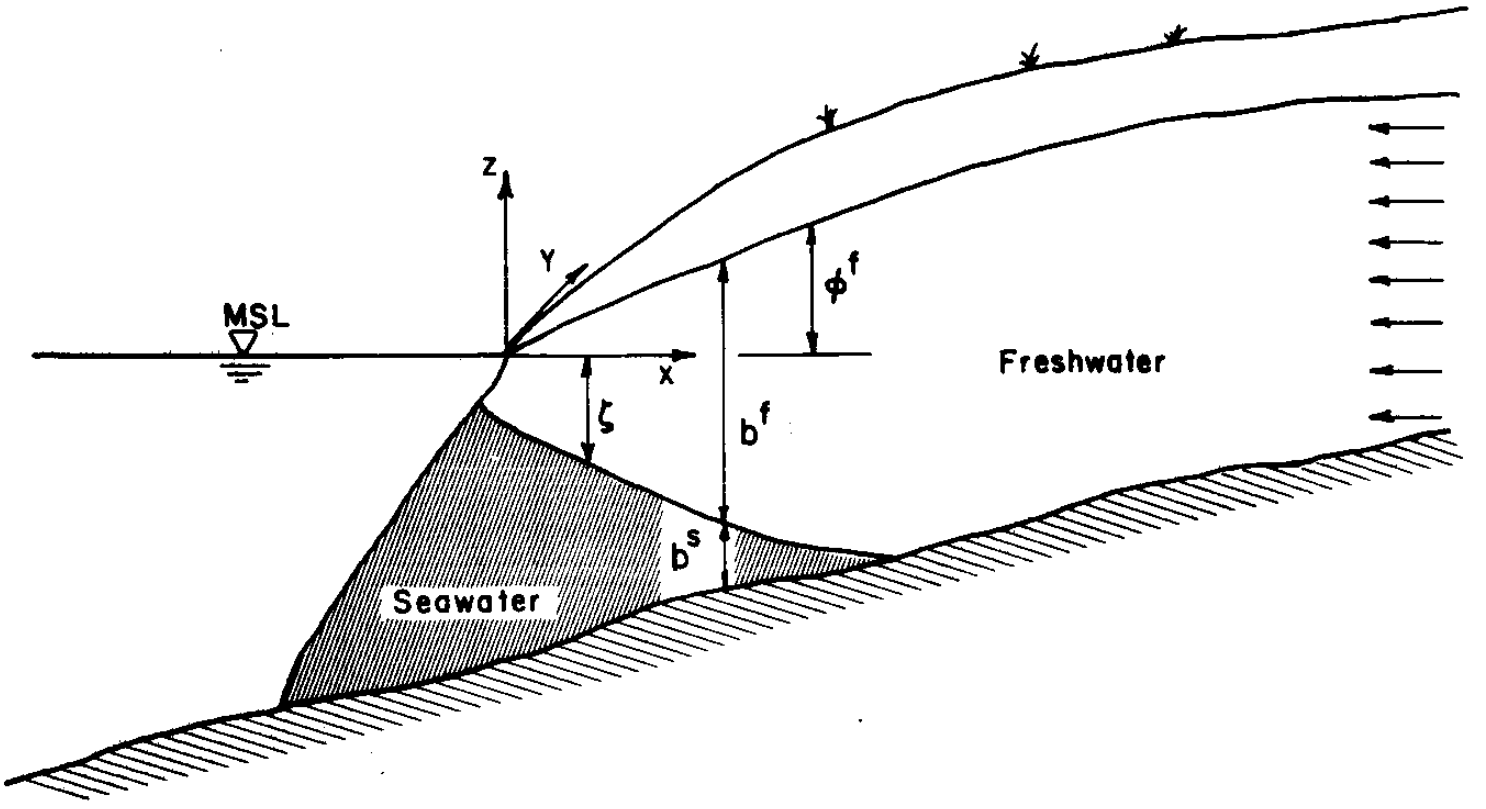


Figure A.2 - Typical Cross Section of a Coastal Phreatic Aquifer

which is the governing equation for the flow in a confined aquifer with a moving interface at the bottom.

The next step in the generalization of Eq. A.9 is to assume that one of the confining boundaries, for instance the top one, is semi-pervious. If it is also assumed that there is no storage in this semi-pervious layer, the second term of Eq. A.8 is represented by (see, e.g., Bear, 1972):

$$\underline{q}\Big|_{z_1} \nabla(z-z_1) = -\frac{K'}{b'}(\phi' - \phi) \qquad (A.10)$$

where: K' = vertical permeability of the semi-pervious or leaky layer,

b' = thickness of the leaky layer, (see Fig. A.3)

ϕ' = piezometric head on the aquifer above the leaky layer.

Hence Eq. A.9 becomes:

$$\frac{\partial}{\partial x}\left(K_x b \frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y b \frac{\partial \phi}{\partial y}\right) - \frac{K'}{b'}(\phi' - \phi) = S_s b \frac{\partial \phi}{\partial t} - n \frac{\partial \zeta}{\partial t} \qquad (A.11)$$

The general governing equation for the freshwater flow in a confined aquifer, with or without leakage, and with a moving freshwater/seawater interface is obtained by introducing into Eq. A.11 terms for a leaky layer at the bottom and a source/sink flux (e.g., recharge/pumping wells), and redefining the variable ζ. The resulting equation is:

$$\underbrace{\frac{\partial}{\partial x}\left(K_x b \frac{\partial \phi}{\partial x}\right)}_{(1)} + \underbrace{\frac{\partial}{\partial y}\left(K_y b \frac{\partial \phi}{\partial y}\right)}_{(2)} + \underbrace{\frac{K'_1}{b'_1}(\phi'_1 - \phi)}_{(3)} + \underbrace{\frac{K'_2}{b'_2}(\phi'_2 - \phi)}_{(4)} + \underbrace{q}_{(5)} =$$

$$= \underbrace{S_s b \frac{\partial \phi}{\partial t}}_{(6)} - \underbrace{n \frac{\partial \zeta}{\partial t}}_{(7)} \qquad (A.12)$$

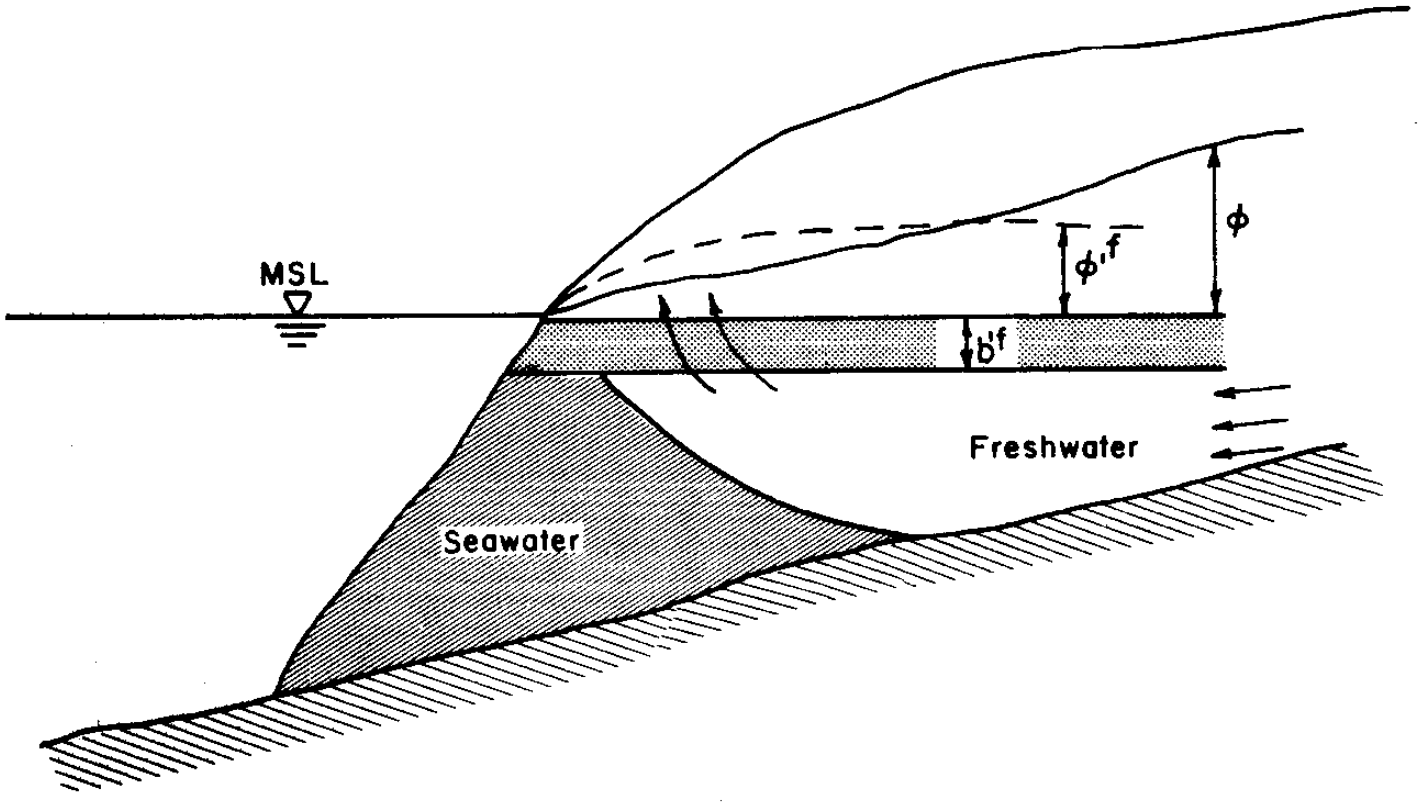


Figure A.3 - Typical Cross Section of a Leaky Coastal aquifer

where: K_1' = vertical permeability of the top leaky layer of thickness b_1' ($K_1' = 0$ if top layer is impervious);

K_2' = vertical permeability of bottom leaky layer of thickness b_2' ($K_2' = 0$ if bottom layer is impervious or if it is the interface);

ζ = interface depth below mean sea level (MSL), as defined in Figure A.3. Interface "toe" is defined by the intersection of the interface with the bottom boundary. Inland of "toe" ζ equals the depth of the bottom boundary below MSL;

$q = Q\delta(x-x_i)(y-y_i)$ = source/sink flow of a well located at point (x_i,y_i), and with strength Q [L/T];

δ = Dirac delta function.

A similar equation can be written for the seawater phase. To distinguish the two equations a superscript ^{f} or ^{s} is used to represent the freshwater and seawater quantities, respectively.

The physical meaning of the individual terms in Eq. A.12 is:

(1),(2) = Darcy's flux in the x and y direction, respectively;

(3),(4) = leaky flux across the top and bottom leaky layers, respectively. These terms vanish for impervious confining boundaries. Term (4) also vanishes if the lower boundary of the freshwater zone is the interface. Similarly, term (3) vanishes for the seawater zone if the top boundary is the interface;

(5) = net recharge or withdrawal at location (x_i,y_i);

(6) = change in storage due to variations in the piezometric head;

(7) = change in storage due to movement of the interface.

A.2.2 Phreatic Aquifers

For a phreatic aquifer with a rate of accretion N($\underline{N} = -N\,\underline{\ell}_z$) the kinematic boundary condition of the phreatic surface, s = 0, is (Bear, 1972):

$$\frac{ds}{dt} = \frac{\partial s}{\partial t} + \underline{v}_s \cdot \nabla s = 0 \qquad \text{(A.13)}$$

where $\underline{v}_s$ = velocity of propagation of the free surface. The continuity equation at this surface is:

$$(\underline{q} - \underline{N}) \cdot \nabla s = n\,\underline{v}_s \cdot \nabla s$$

Therefore:

$$\underline{q} \cdot \nabla S = \underline{N} \cdot \nabla s - n \frac{\partial s}{\partial t} \qquad \text{(A.14)}$$

At the free surface $s = \phi = z_1$, because the pressure p = 0, and Eq. A.14 is equivalent to

$$\underline{q}\Big|_{z_1} \cdot \nabla z = \underline{N} \cdot \nabla z - n \frac{\partial \phi}{\partial t} \qquad \text{(A.15)}$$

Introducing Eq. A.15 in Eq. A.3 and taking $z_1 = \phi$ one can write:

$$\nabla' \int_{z_2}^{\phi} \underline{q}'\,dz - \underline{N} \cdot \nabla z + n \frac{\partial \phi}{\partial t} + \underline{q}_{z_2} \cdot \nabla (z - z_2) =$$

$$= -S_s \frac{\partial}{\partial t} \left(\int_{z_2}^{\phi} \phi\, dz \right) + S_s \phi\Big|_{\phi} \cdot \frac{\partial \phi}{\partial t} - S_s\, \phi\Big|_{z_2} \cdot \frac{\partial z_2}{\partial t} \qquad \text{(A.16)}$$

Assuming that the Dupuit approximation is valid, equipotential surfaces are vertical, see Bear (1972), and using Liebnitz' rule on Eq. A. 16 gives:

$$\frac{\partial}{\partial x}\left(K_x b \frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y b \frac{\partial \phi}{\partial y}\right) + N = (S_s b + n)\frac{\partial \phi}{\partial t} \qquad (A.17)$$

which is equivalent to Eq. A.6, but for unconfined flow with accretion. In this equation the saturated thickness b is now a function of the peizometric head ϕ ($b = \phi - z_2$). From Eq. A. 17 one can easily introduce the moving interface and leaky conditions as previously done for the confined aquifer.

A.2.3 General Governing Equations

Using Eqs. A.12 and A.17 and defining the storage coefficient as $S = S_s b$ = elastic storativity, for confined aquifers, and $S = (S_s b + n)$ = elastic storativity plus specific yield, for unconfined aquifers, the general governing equations are written as (see Fig. A.2):

FRESHWATER:

$$\frac{\partial}{\partial x}\left(K_x^f b^f \frac{\partial \phi^f}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y^f b^f \frac{\partial \phi^f}{\partial y}\right) + \frac{K_1'^f}{b_1'^f}\left(\phi_1'^f - \phi^f\right) + \frac{K_2'^f}{b_2'^f}\left(\phi_2'^f - \phi^f\right)$$

$$+ q^f + N = S^f \frac{\partial \phi^f}{\partial t} - n\frac{\partial \zeta}{\partial t} \qquad (A.18)$$

SEAWATER:

$$\frac{\partial}{\partial x}\left(K_x^s b^s \frac{\partial \phi^s}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y^s b^s \frac{\partial \phi^s}{\partial y}\right) + \frac{K_1'^s}{b_1'^s}\left(\phi_1'^s - \phi^s\right) + \frac{K_2'^s}{b_2'^s}\left(\phi_2'^s - \phi^s\right) =$$

$$= S^s \frac{\partial \phi^s}{\partial t} + n \frac{\partial \zeta}{\partial t} \qquad (A.19)$$

The first term on the right-hand side of Eq. A.19 is almost always negligible.

There are two equations, Eqs. A.18 and A.19, and three unknowns, ϕ^f, ϕ^s, ζ. The equation representing the continuity in pressure at the interface makes the system of equations soluble. This equation is

$$p^s = p^f \qquad (A.20)$$

where p^s, p^f are the pressure on the seawater and freshwater sides of the interface respectively. By definition, the piezometric head in both sides of the interface is expressed as:

$$\phi^f = \frac{p^f}{\gamma^f} + \zeta \quad \text{and} \quad \phi^s = \frac{p^s}{\gamma^s} + \zeta \qquad (A.21)$$

assuming the vertical axis with origin at the intersection of the ground with the mean sea level, and oriented upwards, see Fig.A.3. Combining Eq. A.20 and Eq. A.21

$$\gamma^f(\phi^f - \zeta) = \gamma^s(\phi^s - \zeta)$$

or

$$\phi^s = \frac{\gamma^f}{\gamma^s} \phi^f + \frac{\Delta\gamma}{\gamma^s} \zeta \qquad (A.22)$$

where

$$\Delta\gamma = \gamma^s - \gamma^f$$

Another way of writing Eq. A.22 is

$$\zeta = \frac{\gamma^s}{\Delta\gamma} \phi^s - \frac{\gamma^f}{\Delta\gamma} \phi^f \qquad (A.23)$$

Using Eq. A.22 in Eqs. A.18 and A.19, the two resulting governing equations in terms of freshwater piezometric head, ϕ^f, and interface depth, ζ, are:

FRESHWATER:

$$\frac{\partial}{\partial x}\left(K_x^f b^f \frac{\partial \phi^f}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y^f b^f \frac{\partial \phi^f}{\partial y}\right) + \frac{K_1'^f}{b_1'^f}\left(\phi_1'^f - \phi^f\right) + \frac{K_2'^f}{b_2'^f}\left(\phi_2'^f - \phi^f\right)$$

$$+ q^f + N = S^f \frac{\partial \phi^f}{\partial t} - n \frac{\partial \zeta}{\partial t} \qquad \text{(A.24a)}$$

SEAWATER:

$$\frac{\partial}{\partial x}\left[K_x^s b^s \frac{\partial}{\partial x}\left(\frac{\gamma^f}{\gamma^s}\phi^f + \frac{\Delta\gamma}{\gamma^s}\zeta\right)\right] + \frac{\partial}{\partial y}\left[K_y^s b^s \frac{\partial}{\partial y}\left(\frac{\gamma^f}{\gamma^s}\phi^f + \frac{\Delta\gamma}{\gamma^s}\zeta\right)\right]$$

$$+ \frac{K_1'^s}{b_1'^s}\left(\phi_1'^s - \frac{\gamma^f}{\gamma^s}\phi^f - \frac{\Delta\gamma}{\gamma^s}\zeta\right) + \frac{K_2'^s}{b_2'^s}\left(\phi_2'^s - \frac{\gamma^f}{\gamma^s}\phi^f - \frac{\Delta\gamma}{\gamma^s}\right)$$

$$+ q^s = S^s \frac{\gamma^f}{\gamma^s}\frac{\partial \phi^f}{\partial t} + \left(n + S^s \frac{\Delta\gamma}{\gamma^s}\right)\frac{\partial \zeta}{\partial t} \qquad \text{(A.24b)}$$

If instead of Eq. A.22 we use Eq. A.23, the resulting two governing equations are expressed in terms of both freshwater and seawater piezometric heads, ϕ^f and ϕ^s:

FRESHWATER:

$$\frac{\partial}{\partial x}\left(K_x^f b^f \frac{\partial \phi^f}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y^f b^f \frac{\partial \phi^f}{\partial y}\right) + \frac{K_1'^f}{b_1'^f}\left(\phi_1'^f - \phi^f\right) + \frac{K_2'^f}{b_2'^f}\left(\phi_2^f - \phi^f\right)$$

$$+ q^f + N = \left(S^f + n\frac{\gamma^f}{\Delta\gamma}\right)\frac{\partial \phi^f}{\partial t} - n\frac{\gamma^s}{\Delta\gamma}\frac{\partial \phi^s}{\partial t} \qquad \text{(A.25a)}$$

SEAWATER:

$$\frac{\partial}{\partial x}\left(K_x^s b^s \frac{\partial \phi^s}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y^s b^s \frac{\partial \phi^s}{\partial y}\right) + \frac{K_1'^s}{b_1'^s}\left(\phi_1'^s - \phi^s\right) + \frac{K_2'^s}{b_2'^s}\left(\phi_2'^s - \phi^s\right)$$

$$+ q^s = \left(S^s + n\frac{\gamma^s}{\Delta\gamma}\right)\frac{\partial \phi^s}{\partial t} - n\frac{\gamma^f}{\Delta\gamma}\frac{\partial \phi^f}{\partial t} \qquad \text{(A.25b)}$$

Similar derivations can be found in Pinder and Page (1977) and Bear (1979).

A.3 Tensor Notation of the Preferred Governing Equation

For the SWIM code the form of the governing equations expressed by Eq. A.25 was preferred because it minimizes the storage required for the coefficient matrices, and because it simplifies the computation involved with the evaluation of the storage matrix.

At the beginning of the derivation of the governing equations in Section A.2 principal directions of the permeability tensor were assumed for simplicity of the derivation. Generalizing now to any arbitrary orientation of the axis x_1 and x_2, Eq. A.25 is written in tensor notation* as:

---

*The repetition of indexes in a term represents a summation. For example:

$$\frac{\partial}{\partial x_i}\left(K^f_{x_i x_j} b^f \frac{\partial \phi^f}{\partial x_j}\right) = \frac{\partial}{\partial x_1}\left(K^f_{x_1 x_1} b^f \frac{\partial \phi^f}{\partial x_1}\right) + \frac{\partial}{\partial x_1}\left(K^f_{x_1 x_2} b^f \frac{\partial \phi^f}{\partial x_2}\right) +$$

$$+ \frac{\partial}{\partial x_2}\left(K^f_{x_2 x_1} b^f \frac{\partial \phi^f}{\partial x_1}\right) + \frac{\partial}{\partial x_2}\left(K^f_{x_2 x_2} b^f \frac{\partial \phi^f}{\partial x_2}\right)$$

FRESHWATER:

$$\frac{\partial}{\partial x_i}\left(K^f_{x_i x_j}\, b^f \frac{\partial \phi^f}{\partial x_j}\right) + \frac{K'^f_m}{b'^f_m}\left(\phi'^f_m - \phi^f\right) + q^f + N =$$

$$= \left(S^f + n\frac{\gamma^f}{\Delta\gamma}\right)\frac{\partial \phi^f}{\partial t} - n\frac{\gamma^s}{\Delta\gamma}\frac{\partial \phi^s}{\partial t} \qquad i,j,m=1,2 \qquad \text{(A.26a)}$$

SEAWATER:

$$\frac{\partial}{\partial x_i}\left(K^s_{x_i x_j}\, b^s \frac{\partial \phi^s}{\partial x_j}\right) + \frac{K'^s_m}{b'^s_m}\left(\phi'^s_m - \phi^s\right) + q^s =$$

$$= \left(S^s + n\frac{\gamma^s}{\Delta\gamma}\right)\frac{\partial \phi^s}{\partial t} - n\frac{\gamma^f}{\Delta\gamma}\frac{\partial \phi^f}{\partial t} \qquad i,j,m=1,2 \qquad \text{(A.26b)}$$

These two equations can be interpreted as two degrees of freedom of the piezomteric head state variable:

$$\underline{\Phi} = \begin{bmatrix} \phi^t \\ \phi^s \end{bmatrix} = \underline{\Phi}\,(x,y,t)$$

Thus, Eq. A.26 can be written in matrix notation as:

$$\underline{\underline{B}}\;\underline{\underline{T}}\;\underline{\underline{B}}^T\,\underline{\Phi} + \underline{\underline{K}}'_m\left(\underline{\Phi}'_m - \underline{\Phi}\right) + \underline{Q} = \underline{\underline{S}}\frac{\partial \underline{\Phi}}{\partial t} \qquad m=1,2 \qquad \text{(A.27)}$$

where the following matrices ($\underline{\underline{\ }}$) and vectors ($\underline{\ }$) are introduced:

$$\underline{\underline{B}} = \begin{bmatrix} \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} \end{bmatrix}$$

$$\underline{\underline{T}} = \begin{bmatrix} K^f_{x_1x_1} b^f & K^f_{x_1x_2} b^f & 0 & 0 \\ K^f_{x_2x_1} b^f & K^f_{x_2x_2} b^f & 0 & 0 \\ 0 & 0 & K^s_{x_1x_2} b^s & K^s_{x_1x_2} b^s \\ 0 & 0 & K^s_{x_2x_1} b^s & K^s_{x_2x_2} b^s \end{bmatrix}$$

$$\underline{\underline{K}}'_m = \begin{bmatrix} K'^f_m / b'^f_m & 0 \\ 0 & K'^s_m / b'^s_m \end{bmatrix} \qquad m=1,2$$

$$\underline{\phi}'_m = \begin{bmatrix} \phi'^f_m \\ \phi'^s_m \end{bmatrix} \qquad m=1,2$$

$$\underline{Q} = \begin{bmatrix} q^f + N \\ q^s \end{bmatrix}$$

$$\underline{\underline{S}} = \begin{bmatrix} S^f + n\frac{\gamma^f}{\Delta\gamma} & -n\frac{\gamma^s}{\Delta\gamma} \\ -n\frac{\gamma^f}{\Delta\gamma} & S^s + n\frac{\gamma^s}{\Delta\gamma} \end{bmatrix}$$

A superscript ^{T} represents a transpose quantity.

In order to clarify this expression for readers less familiar with tensor algebra, the remaining portion of this section presents a step by step precedure expanding the simplified Eq. A.27 to the full extent of

Eq. A.26. Examining the first term on the left-hand side of Eq. A.27, matrix multiplications will be performed one at a time:

$$\underline{\underline{B}}^T \underline{\Phi} = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 \\ \frac{\partial}{\partial x_2} & 0 \\ 0 & \frac{\partial}{\partial x_1} \\ 0 & \frac{\partial}{\partial x_2} \end{bmatrix} \cdot \begin{bmatrix} \phi^f \\ \phi^s \end{bmatrix} = \begin{bmatrix} \frac{\partial \phi^f}{\partial x_1} \\ \frac{\partial \phi^f}{\partial x_2} \\ \frac{\partial \phi^s}{\partial x_1} \\ \frac{\partial \phi^s}{\partial x_2} \end{bmatrix}$$

Then:

$$\underline{\underline{T}}\,\underline{\underline{B}}^T \underline{\Phi} = \underline{\underline{T}}(\underline{\underline{B}}^T \underline{\Phi}) = \begin{bmatrix} K^f_{x_1x_1} b^f & K^f_{x_1x_2} b^f & 0 & 0 \\ K^f_{x_2x_1} b^f & K^f_{x_2x_2} b^f & 0 & 0 \\ 0 & 0 & K^s_{x_1x_1} b^s & K^s_{x_1x_2} b^s \\ 0 & 0 & K^s_{x_2x_1} b^s & K^s_{x_2x_2} b^s \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \phi^f}{\partial x_1} \\ \frac{\partial \phi^f}{\partial x_1} \\ \frac{\partial \phi^s}{\partial x_1} \\ \frac{\partial \phi^s}{\partial x_1} \end{bmatrix}$$

$$= \begin{bmatrix} K^f_{x_1x_1} b^f \frac{\partial \phi^f}{\partial x_1} + K^f_{x_1x_2} b^f \frac{\partial \phi^f}{\partial x_2} \\ K^f_{x_2x_1} b^f \frac{\partial \phi^f}{\partial x_1} + K^f_{x_2x_2} b^f \frac{\partial \phi^f}{\partial x_2} \\ K^s_{x_1x_1} b^s \frac{\partial \phi^s}{\partial x_1} + K^s_{x_1x_2} b^s \frac{\partial \phi^s}{\partial x_2} \\ K^s_{x_2x_1} b^s \frac{\partial \phi^s}{\partial x_1} + K^s_{x_2x_2} b^s \frac{\partial \phi^s}{\partial x_2} \end{bmatrix}$$

Finally:

$$\underline{\underline{B}}\ \underline{\underline{T}}\ \underline{\underline{B}}^T\ \underline{\phi} = \underline{\underline{B}}(\underline{\underline{T}}\ \underline{\underline{B}}^T\ \underline{\phi}) =$$

$$= \begin{bmatrix} \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} \end{bmatrix} \cdot \begin{bmatrix} K^f_{x_1x_1} b^f \frac{\partial \phi^f}{\partial x_1} + K^f_{x_1x_2} b^f \frac{\partial \phi^f}{\partial x_2} \\ K^f_{x_2x_1} b^f \frac{\partial \phi^f}{\partial x_1} + K^f_{x_2x_2} b^f \frac{\partial \phi^f}{\partial x_2} \\ K^s_{x_1x_1} b^s \frac{\partial \phi^s}{\partial x_1} + K^s_{x_1x_2} b^s \frac{\partial \phi^s}{\partial x_2} \\ K^s_{x_2x_1} b^s \frac{\partial \phi^s}{\partial x_1} + K^s_{x_2x_2} b^s \frac{\partial \phi^s}{\partial x_2} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial}{\partial x_1}\left(K^f_{x_1x_1} b^f \frac{\partial \phi^f}{\partial x_1}\right) + \frac{\partial}{\partial x_1}\left(K^f_{x_1x_2} b^f \frac{\partial \phi^f}{\partial x_2}\right) + \\ \vdots \\ + \frac{\partial}{\partial x_2}\left(K^f_{x_2x_1} b^f \frac{\partial \phi^f}{\partial x_1}\right) + \frac{\partial \phi}{\partial x_2}\left(K^f_{x_2x_2} b^f \frac{\partial \phi^f}{\partial x_2}\right) \\ \vdots \end{bmatrix}$$

$$\begin{bmatrix} \vdots \\ \frac{\partial}{\partial x_1}\left(K^s_{x_1x_1} b^s \frac{\partial \phi^s}{\partial x_1}\right) + \frac{\partial}{\partial x_1}\left(K^s_{x_1x_2} b^s \frac{\partial \phi^s}{\partial x_2}\right) \\ + \frac{\partial}{\partial x_2}\left(K^s_{x_2x_1} b^s \frac{\partial \phi^s}{\partial x_1}\right) + \frac{\partial}{\partial x_2}\left(K^s_{x_2x_2} b^s \frac{\partial \phi^s}{\partial x_2}\right) \\ \vdots \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial}{\partial x_i}\left(K^f_{x_ix_j} b^f \frac{\partial \phi^f}{\partial x_j}\right) \\ \\ \frac{\partial}{\partial x_i}\left(K^s_{x_ix_j} b^s \frac{\partial \phi^s}{\partial x_j}\right) \end{bmatrix} \qquad (A.28)$$

The second term on the left-hand side of Eq. A.27

$$\underline{\underline{K}}'_m \left(\underline{\phi}'_m - \underline{\phi}\right) = \begin{bmatrix} K'^f_m/b^f_m & 0 \\ 0 & K'^s_m/b'^s_m \end{bmatrix} \cdot \begin{bmatrix} \phi'^f_m - \phi^f \\ \phi'^s_m - \phi^s \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{K'^f_m}{b'^f_m}\left(\phi'^f_m - \phi^t\right) \\ \\ \frac{K'^s_m}{b'^s_m}\left(\phi'^s_m - \phi^s\right) \end{bmatrix} \qquad m=1,2 \qquad (A.29)$$

The last term on the left-hand side is given by definition.

$$\underline{Q} = \begin{bmatrix} q^f + N \\ q^s \end{bmatrix} \qquad (A.30)$$

The right-hand side is:

$$\underline{\underline{S}}\,\frac{\partial \underline{\phi}}{\partial t} = \begin{bmatrix} S^f + n\frac{\gamma^f}{\Delta\gamma} & -n\frac{\gamma^s}{\Delta\gamma} \\ -n\frac{\gamma^f}{\Delta\gamma} & S^s + n\frac{\gamma^s}{\Delta\gamma} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \phi^f}{\partial t} \\ \frac{\partial \phi^s}{\partial t} \end{bmatrix} =$$

$$= \begin{bmatrix} (S^f + n\frac{\gamma^f}{\Delta\gamma})\frac{\partial \phi^f}{\partial t} - n\frac{\gamma^s}{\Delta\gamma}\frac{\partial \phi^s}{\partial t} \\ -n\frac{\gamma^f}{\Delta\gamma}\frac{\partial \phi^f}{\partial t} + (S^s + n\frac{\gamma^s}{\Delta\gamma})\frac{\partial \phi^s}{\partial t} \end{bmatrix} \qquad \text{(A.31)}$$

Adding Eq. A.28 through A.31 the equivalent to Eq. A.27 is obtained:

$$\begin{bmatrix} \frac{\partial}{\partial x_i}(K^f_{x_i x_j} b^f \frac{\partial \phi^f}{\partial x_j}) + \frac{K'^f_m}{b'^f_m}(\phi'^f_m - \phi^f) + \underline{q}^f + N \\ \frac{\partial}{\partial x_i}(K^s_{x_i x_j} b^s \frac{\partial \phi^s}{\partial x_j}) + \frac{K'^s_m}{b'^s_m}(\phi'^s_m - \phi^s) + q^s \end{bmatrix} = \qquad \text{(A.32)}$$

$$= \begin{bmatrix} (S^f + n\frac{\gamma^f}{\Delta\gamma})\frac{\partial \phi f}{\partial t} - n\frac{\gamma^s}{\Delta\gamma}\frac{\partial \phi^s}{\partial f} \\ -n\frac{\gamma f}{\Delta\gamma}\frac{\partial \phi f}{\partial t} + (S^s + n\frac{\gamma^s}{\Delta\gamma})\frac{\partial \phi^s}{\partial t} \end{bmatrix} \quad m=1,2 \qquad \text{(A.32)}$$

A.4 Summary

The general differential governing equations for the groundwater flow in aquifers with seawater intrusion have been derived, in terms of freshwater piezometric head and interface depth, and in terms of freshwater and seawater piezometric heads. The short tensor form of this last equation has also been developed:

$$\underline{\underline{B}}\ \underline{\underline{T}}\ \underline{\underline{B}}^{T}\ \underline{\Phi} + \underline{\underline{K}}'_{m}(\underline{\Phi}_{m} - \underline{\Phi}) + \underline{Q} = \underline{\underline{S}}\ \frac{\partial \underline{\Phi}}{\partial t}\Bigg|_{m=1,2} \qquad (A.27)$$

In the derivation of these equations the following assumptions have been considered:

1) Immiscible interface separating the two fluids;
2) Darcy's Law is valid;
3) Specific storage and horizontal permeability variations in the vertical direction are negligible;
4) Dupuit approximation for phreatic aquifers and essentially horizontal flow for confined aquifers. Vertical flow considered only in the case of leakage from adjacent aquifers or in case of accretion in phreatic aquifers.

Inherent to the physical process it has also been assumed that both freshwater and seawater have constant density, and are homogeneous and isotropic fluids filling completely all the voids in the saturated zone of the porous media.

Appendix B

CHECK OF THE POSITIVE SEMI-DEFINITENESS OF THE EFFECTIVE CONDUCTIVITY MATRIX USED IN SWIM

This appendix discusses the positive semi-definiteness of the effective conductivity matrix defined in Table 3.1, which is required to solve the system of equations using the $\underline{\underline{L}}^T \underline{\underline{D}} \underline{\underline{L}}$ method. A matrix is said to be positive definite if all eigenvalues are positive. For ease of calculation the analysis presented here is in terms of elemental matrices and local coordinates. If the typical element matrix is positive definite, so too the global matrix.

The effective conductivity matrix $\underline{\underline{K}}^*$ as defined in Chapter 3 is formed by the summation of the conductivity matrix $\underline{\underline{T}}$, storage matrix divided by the time step $\underline{\underline{M}} = \underline{\underline{C}}/\Delta t$ and the third type boundary condition matrices $\underline{\underline{L}}_1$ and $\underline{\underline{L}}_2$. Since these last two matrices have all positive elements they are not considered for this analysis of the eigenvalues of the effective conductivity matrix--in any case they slightly strengthen the positive definiteness of the global matrix. In order to simplify the calculations and without loss of generality it is assumed that both freshwater and seawater transmissivity and hydraulic conductivity are the same, that elastic storativity is negligible, and that $\gamma = \gamma^f = 1.0\ g/cm^3$, $\gamma^s = \gamma + \Delta\gamma$ and $\Delta\gamma = 0.03$. Under these assumptions the element matrices for a 4 node element are defined as:

$$\underline{\underline{T}} = \int_{A^e} \underline{\underline{B}}^T \underline{\underline{T}} \underline{\underline{B}}\, dA = Kb \int_{A^e} \underline{\underline{B}}^T \underline{\underline{B}}\, dA \qquad (B.1)$$

$$\underline{\underline{M}} = \frac{\underline{\underline{C}}}{\Delta t} = \frac{1}{\Delta t} \int_{A^e} \underline{\underline{H}}^T \, \underline{\underline{S}} \, \underline{\underline{H}} \, dA \qquad (B.2)$$

where:

$$\underline{\underline{S}} = \begin{bmatrix} n\frac{\gamma}{\Delta\gamma} & -n\frac{\gamma+\Delta\gamma}{\Delta\gamma} \\ -n\frac{\gamma+\Delta\gamma}{\Delta\gamma} & n\frac{\gamma+\Delta\gamma}{\Delta\gamma} \end{bmatrix} =$$

$$= \frac{n\gamma}{\Delta\gamma} \cdot \begin{bmatrix} 1 & -\varepsilon \\ -\varepsilon & \varepsilon \end{bmatrix}$$

$$\varepsilon = 1 + \frac{\Delta\gamma}{\gamma}$$

$$\underline{\underline{B}} = \frac{1}{4} \cdot \begin{bmatrix} (1+s) & 0 & -(1+s) & 0 & -(1-s) & 0 & (1-s) & 0 \\ (1+r) & 0 & (1-r) & 0 & -(1-r) & 0 & -(1+r) & 0 \\ 0 & (1+s) & 0 & -(1+s) & 0 & -(1-s) & 0 & (1-s) \\ 0 & (1+r) & 0 & (1-r) & 0 & -(1-r) & 0 & -(1+r) \end{bmatrix}$$

$$\underline{\underline{H}} = \frac{1}{4} \cdot \begin{bmatrix} (1+r)(1+s) & 0 & (1-r)(1+s) & 0 \\ 0 & (1+r)(1+s) & 0 & (1-r)(1+s) \end{bmatrix}$$

$$\begin{bmatrix} (1-r)(1-s) & 0 & (1+r)(1-s) & 0 \\ 0 & (1-r)(1-s) & 0 & (1+r)(1-s) \end{bmatrix}$$

r and s are the local coordinate system.

Performing the matrix multiplications in Eq. B.1 and B.2 and the integration one obtains:

$$\underline{\underline{T}} = \frac{Kb}{12} \cdot \begin{bmatrix} 4 & 0 & -1 & 0 & -2 & 0 & -1 & 0 \\ & 4 & 0 & -1 & 0 & -2 & 0 & -1 \\ & & 4 & 0 & -1 & 0 & -2 & 0 \\ & & & 4 & 0 & -1 & 0 & -2 \\ & & & & 4 & 0 & -1 & 0 \\ & & & & & 4 & 0 & -1 \\ & & & & & & 4 & 0 \\ & & & & & & & 4 \end{bmatrix} \qquad (B.3)$$

$$\underline{\underline{M}} = \frac{1}{9} \frac{n\gamma}{\Delta\gamma\Delta t} \cdot \begin{bmatrix} 4 & -4\varepsilon & 2 & -2\varepsilon & 1 & -\varepsilon & 2 & -2\varepsilon \\ & 4\varepsilon & -2\varepsilon & 2\varepsilon & -\varepsilon & \varepsilon & -2\varepsilon & 2\varepsilon \\ & & 4 & -4\varepsilon & 2 & -2\varepsilon & 1 & -\varepsilon \\ & & & 4\varepsilon & -2\varepsilon & 2\varepsilon & -\varepsilon & \varepsilon \\ & & & & 4 & -4\varepsilon & 2 & -2\varepsilon \\ & & & & & 4\varepsilon & -2\varepsilon & 2\varepsilon \\ & & & & & & 4 & -4\varepsilon \\ & & & & & & & 4\varepsilon \end{bmatrix} \qquad (B.4)$$

Adding these two matrices and defining $X = \frac{n\gamma}{\Delta\gamma\Delta t}$

$$\underline{\underline{K}}^* = \frac{1}{36} \cdot \begin{bmatrix} 16X+12Kb & -16X\varepsilon & 8X-3Kb & -8X\varepsilon & 4X-6Kb & -4X\varepsilon & 8X-3Kb & -8X\varepsilon \\ & 16X\varepsilon+12Kb & -8X\varepsilon & 8X\varepsilon-3Kb & -4X\varepsilon & 4X\varepsilon-6Kn & -8X\varepsilon & 8X\varepsilon-3Kb \\ & & 16X+12Kb & -16X\varepsilon & 8X-3Kb & -8X\varepsilon & 4X-6Kb & -4X\varepsilon \\ & & & 16X\varepsilon+12Kb & -8X\varepsilon & 8X\varepsilon-3Kb & -4X\varepsilon & 4X\varepsilon-6Kb \\ & & & & 16X+12Kb & -16X\varepsilon & 8X-3Kb & -8X\varepsilon \\ & & & & & 16X\varepsilon+12Kb & -8X\varepsilon & 8X\varepsilon-3Kb \\ & & & & & & 16X+12Kb & -16X\varepsilon \\ & & & & & & & 16X\varepsilon+12Kb \end{bmatrix} \qquad (B-5)$$

The calculations presented below for $\underline{\underline{K}}^*$ ignore the constant factor 1/36 in the previous equation.

The eignevalues for matrix $\underline{\underline{I}}$, Eq. B.3, assuming Kb = 1, are:

$$\lambda_1 = \lambda_2 = 0.0$$

$$\lambda_3 = \lambda_4 = 12.0$$

$$\lambda_5 = \lambda_6 = \lambda_7 = \lambda_8 = 18.0$$

The fact that two of the eigenvalues are zero means that the $\underline{\underline{I}}$ matrix cannot represent the so called rigid body modes found in structures. That is, the solution needs to be anchored for both freshwater and seawater phase by imposing first (or third) type boundary conditions. This can be achieved by adding large numbers, say 10^6, to two of the diagonal terms. Doing so leads to the eigenvalues:

$$\lambda_1 = \lambda_2 = 3.804$$

$$\lambda_3 = \lambda_4 = 14.196$$

$$\lambda_5 = \lambda_6 = 18.000$$

$$\lambda_7 = \lambda_8 = 10^6$$

This means that for any value of permeability, which is always positive, matrix $\underline{\underline{I}}$ is always positive definite, and the solution of a steady state problem is always defined.

So far the eigenvalues have been computed analytically due to the simplicity of the matrix; however, from this point on the eigenvalues will be computed using numerical methods. Therefore, some roundoff and

truncation errors may result from the single precision digits used in the computer packages employed.

Assuming n = Δt and neglecting the factor 1/9 in Eq. B.3 the eigenvalues for matrix $\underline{\underline{M}} = \underline{\underline{C}}/\Delta t$ (Eq. B.4) are:

$\lambda_1 = \lambda_2 = -6.043$

$\lambda_2 = -2.014$

$\lambda_4 = -1.813$

$\lambda_5 = 2.726$

$\lambda_6 = \lambda_7 = 818.0$

$\lambda_8 = 245.4$

The four negative eigenvalues are not surprising given the type of matrix involved.

The results of the eigenvalues of matrix $\underline{\underline{K}}^*$ for different values of the dimensionless term $\tau = Kb\Delta t/nL^2$ are presented in Table B.1, where L=1 is the half length of an element side in local coordinates. τ=1 is an extreme case of a very low transmissivity and very small time step which has no real meaning but shows that the matrix $\underline{\underline{K}}^*$ is not always positive definite. τ=5, 10 are more realistic values and they can be used to test some experimental results, however these values of τ are still too small compared with common field values. $\underline{\underline{K}}^*$ is now positive definite and its positive definiteness increases with τ. τ = 15000 can occur for the case of Kb = 100 m^2/day, n = 0.2 and Δt = 30 days, which are common for a field problem, $\tau \simeq 17 \times 10^4$ is obtained when larger values for transmissivity and time step are used, Δt = 1 year for example. In these cases the transmissivity matrix $\underline{\underline{T}}$ dominates the

Table B.1

Eigenvalues of $\underline{\underline{K}}^*$ for Different Values of τ

eigenvalues	τ				
	1	5	10	15000	17×10^4
γ_1	-1.07	.92	23.65	379.6	1.76×10^3
γ_2	1.07	13.5	138.0	392.9	3.76×10^3
γ_3	12.05	16.95	174.2	1420.	1.42×10^4
γ_4	457.8	102.8	580.7	1423.	1.42×10^4
γ_5	836.0	181.6	998.1	1800.	1.8×10^4
γ_6	2014.	406.3	2053.	1805.	1.8×10^4
γ_7	10^6	10^6	10^6	10^6	10^6
γ_8	10^6	10^6	10^6	10^6	10^6

storage matrix $\underline{\underline{M}}$ and one can see that the eigenvalues of $\underline{\underline{K}}^*$ are the eigenvalues of $\underline{\underline{\ }}$ multiplied by Kb.

For all pratical purposes the effective conductivity matrix $\underline{\underline{K}}^*$ is positive definite and the use of any Gauss elimination technique to solve the system of equations is appropriated. In some cases, however, the the non-positive definiteness of the effective conductivity matrix plays a role, and $\underline{\underline{K}}^*$ cannot be decomposed and solved using Gauss elimination. In these cases the storage matrix can be lumped (an option available in SWIM, see sections 3.6, 4.3.4.6, and 5.7) and the solution can proceed.

APPENDIX C

INITIAL AND BOUNDARY CONDITIONS USED IN THE 1-D GRAVITATIONAL SEGREGATION PROBLEM DESCRIBED IN CHAPTER 4

Initial Conditions *

x (m)	ϕ^f (m)	ζ (m)	ϕ^s (m)
< -16	5.125	-10.0	4.875
-12	5.12305	- 8.75	4.90430
- 8	5.11718	- 7.5	4.92968
- 4	5.10742	- 6.25	4.95117
0	5.09375	- 5.0	4.96875
4	5.07617	-3.75	4.98242
8	5.05469	-2.5	4.99219
12	5.02930	-1.25	4.99805
>16	5.0	0.0	5.0

ϕ^f freshwater piezometric head

ζ interface depth

ϕ^s seawater piezometric head calculated using Eq. 3.3 that is:

$$\phi^s = \Delta\gamma\zeta + \gamma^f\phi^f$$

$\gamma^f = 1.0 g/cm^3$

$\Delta\gamma = 0.025 g/cm^3$

---

* See Figure C.1 .

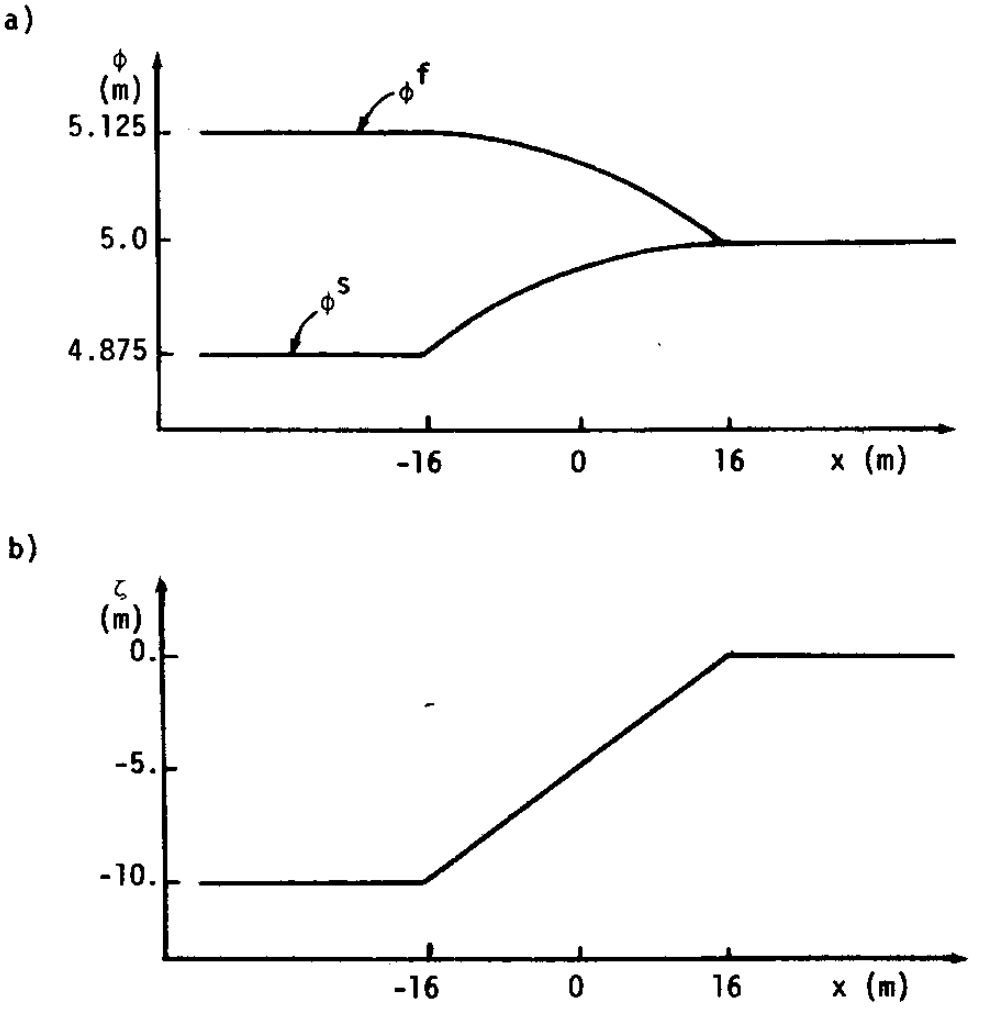


Figure C.1 - Initial Conditions Used in the Gravitational Segregation Problem. a) Piezometric Heads; b) Interface Elevation.

Boundary Conditions

$\phi^f = 5.125$ m

$\phi^s = 4.875$ m

at x = a

a = -68 m for the simulation described in Section 4.3.3 using a longer grid (136 m)

a = -50 m for the simulation described in Section 4.3.4 using a shorter grid (100 m)